

Evaluation of Ramp Metering Algorithms Using Microsimulation

By

© 2018

Mohammad Khalid Shehada

B.Sc. Civil Engineering, Jordan University of Science and Technology, 2015

Submitted to the graduate degree program in Civil, Environmental, and Architectural Engineering and the Graduate Faculty of the University of Kansas in partial fulfillment of the requirements

for the degree of Master of Science in Civil Engineering.

Chair: Dr. Alexandra Kondyli

Dr. Steven D. Schrock

Dr. Thomas E. Mulinazzi

Date Defended: 1 March 2018

The thesis committee for Mohammad Khalid Shehada certifies that
this is the approved version of the following thesis:

Evaluation of Ramp Metering Algorithms Using Microsimulation

Chair: Dr. Alexandra Kondyli

Date Approved: 6 March 2018

ABSTRACT

The goal of the thesis was to determine the effectiveness of implementing different ramp metering strategies along I-35 in Kansas City using microsimulation analysis. Ramp metering enhances traffic conditions on the mainline by restricting the accessibility of the on-ramp traffic. Traffic data for one year (04/01/2016 to 03/31/2017) during the peak period were used to evaluate the performance of the facility before and after implementing known ramp metering strategies. The evaluation was done using the VISSIM microsimulation software.

The locations of the metered junctions along I-35 were obtained from the Kansas Department of Transportation (KDOT), since KDOT installed ramp meters at these locations in 2017. Four ramp meters were located at the southbound direction and two at the northbound direction. In this thesis only the I-35 southbound movements were evaluated, as the meters at the northbound direction were placed primarily for safety purposes. The I-35 southbound corridor starts from Cambridge Dr. in the north and ends at 75th St. in the south. The ramp meters are located at the 7th St., Southwest Blvd., 18th St. Expressway, and 67th St. on-ramps.

Currently, KDOT is implementing a speed-based algorithm, the details of which are unknown since the exact algorithm is proprietary. As such, for the purposes of this thesis a review of the literature was conducted to identify possible ramp metering algorithms to evaluate, and it was decided to use one localized and one system-wide ramp metering algorithm. The selected localized ramp metering algorithm is the ALINEA (Papageorgiou et al., 1991). ALINEA is an occupancy-based ramp metering algorithm that operates to maintain the occupancy in the freeway at the congestion location close to the critical occupancy that corresponds to maximum throughput. The selected system-wide ramp metering algorithm is HERO (Papamichail and Papageorgiou 2008). HERO uses ALINEA as its base algorithm, and uses a master/slave protocol. These two

ramp metering algorithms, as well as the No Control scenario were evaluated considering various performance measures obtained through microsimulation.

Traffic data were obtained from the KC Scout portal. The data obtained were screened for days with adverse weather conditions, traffic incidents, and bad detector data. The remaining data were used to obtain traffic demands and off-ramp relative flows to be used in VISSIM. The three control scenarios (ALINEA, HERO, and No Control) were simulated using 60 demand scenarios. These scenarios were created by averaging the weekday data in each month. Each demand scenario was run four times with different seed numbers to account for variations throughout the week, resulting in a total of 240 simulated days.

The selected performance measures that were used to perform the evaluation were travel time and travel time reliability, speeds, throughput, queue lengths, and congestion duration. The entire facility travel time did not show significant improvement; however, significant travel time improvements were observed at the northern part of the facility. Congestion duration decreased after implementing the ramp metering algorithms at all metered locations except the 67th Street. Mainline spot mean speed at the metered locations also increased. Also, the throughput increased after implementing the ramp metering strategies compared to the No Control scenario.

Overall, ALINEA was found to perform better than HERO; however, ALINEA had longer queues on the on-ramps, spillback percentage to the arterials and waiting times compared to HERO at all the metered locations except at 7th St. on-ramp. This is because in ALINEA a queue flush system was used when the queue length reaches a threshold, while in HERO, a queue control strategy that adapts to queue length was used.

DEDICATION

I would like to dedicate this thesis to my parents, Khalid Hussein Shehada & Abeer Fawzi El-Attar, for supporting me during my journey to obtain my master's degree. There are no thank you words capable of showing the gratitude I have for them for pushing me beyond my limits and raising me up whenever I fall to pursue my dream. I will always be indebted to them and will always love even if my heart stops beating.

ACKNOWLEDGMENTS

I would like to take the opportunity to express my sincere gratitude to my mentor and advisor Dr. Alexandra Kondyli for her continuous support, motivation, and embarking her wisdom to me. She was the manifestation of the perfect advisor to me during my master's program.

I would also like to Dr. Steven D. Schrock and Dr. Thomas E. Mulinazzi, for being part of my committee and for helping me sculpt my thesis to be in better shape.

I would also like to thank my fellow colleagues who are at the same time my friends, Amin Asgharzadeh, Akshit Ahuja, and Hemin Mohammad for helping me find solutions for problems I had in my research and making my journey an enjoyable one.

Contents

1. INTRODUCTION.....	1
1.1. Traffic Flow Theory.....	2
1.2. Ramp Metering Strategies.....	4
1.3. Thesis Objectives.....	5
2. LITERATURE REVIEW	6
2.1. Local Ramp Metering Strategies	6
2.1.1. Demand Capacity (DC)	6
2.1.2. Percent-Occupancy (OCC).....	8
2.1.3. Asservissement Linéaire d'Entrée Autoroutière (ALINEA).....	9
2.1.4. ALINEA Extensions.....	9
2.1.5. Mixed Control	13
2.2. Coordinated Ramp Metering Strategies	14
2.2.1. METALINE.....	14
2.2.2. Zone Algorithm	15
2.2.3. Fuzzy Logic Control (FLC).....	16
2.2.4. HEuristic Ramp-metering CoOrdination (HERO).....	20
2.2.5. System Wide Adaptive Ramp Metering (SWARM)	21

2.3.	Ramp Metering Field Evaluations.....	22
2.3.1.	Monash Freeway, M1 and M3 freeways in Queensland Australia (HERO)	22
2.3.2.	A6W Motorway, Paris – ALINEA.....	23
2.3.3.	I-90 and I-405, Seattle – Fuzzy Logic control	24
2.3.4.	Assessment of FUZZY LOGIC on A12 motorway in Holland.....	24
2.4.	Simulation Evaluations.....	24
2.4.1.	Comparison of HERO, ALINEA, and PI-ALINEA	25
2.4.2.	Comparison of ALINEA, ZONE, and Bottleneck.....	25
2.4.3.	Twin Cities, Minnesota Coordinated Ramp Metering Assessment.....	27
2.5.	Vissim Software.....	27
2.6.	Summary of literature review	29
3.	METHODOLOGY	30
3.1.	Model Development and Calibration	30
3.2.	Selected Ramp Metering Algorithms	31
3.3.	Simulation Process	32
3.4.	Evaluation Results	32
4.	DATA COLLECTION AND SCENARIO DEVELOPMENT	34
4.1.	Facility and Data Overview.....	34
4.2.	Data Formatting and Screening.....	34
4.3.	Data Reduction and Demand Scenario Development.....	35

5. CALIBRATION AND SIMULATION	37
5.1. Modeling the Facility	37
5.2. Vehicle Properties	39
5.3. Calibration.....	40
5.3.1. Connectors Lane Change Distance and Vehicle Routes Decision.....	41
5.3.2. Driving Behavior	42
5.3.3. Calibration Results	43
5.4. Simulation of Control Scenarios.....	46
5.4.1. ALINEA Algorithm	47
5.4.2. HERO Algorithm	49
6. SIMULATION RESULTS	51
6.1. Travel Time	51
6.1.1. Facility Travel Time and Travel Time Reliability	51
6.1.2. First Half Travel Time and Travel Time Reliability	54
6.2. Throughput.....	60
6.3. Congestion Duration	61
6.4. Mean Speed.....	64
6.5. Results Discussion	65
6.6. Limitations.....	67
7. CONCLUSION AND RECOMMENDATIONS	68

8. REFERENCES.....	72
APPENDIX A	77
APPENDIX B	90

1. INTRODUCTION

Over the last half century, traffic demand on freeways and their ramps has increased substantially. This increase is a result of population and auto ownership growth. In the light of this increase, transportation infrastructure started experiencing more congestion, crashes and other transportation problems (Jacobson et al., 2006). To meet the public's needs, improvements to transportation facilities have to be done. Facility improvements can take the form of infrastructure enhancements such as new roadways and additional lanes. However, these enhancements are not always feasible, so other approaches (e.g., traffic optimization) had to be implemented. One popular optimization tool is ramp metering. Ramp metering is a traffic signal controller situated at on-ramps that limits and manages the traffic entering the freeway in order to achieve the optimal capacity of the freeway and prevent or delay the onset of congestion. Ramp metering applications have many goals and benefits such as (Jacobson et al., 2006):

- Increase travel speed, throughput, travel time reliability, and occupancy and maintain traffic operations in uncongested conditions;
- Break up entering platoons which are the main cause of localized congestion; and
- Reduce travel times, emissions, and vehicle crashes.

Ramp metering was first introduced in the early 1960s as a result of studies done on the impact of demand-capacity relationship on traffic operations, such as congestion and safety. In 1963 the first ramp meters were implemented on Chicago's Eisenhower Expressway and were controlled by traffic enforcement officers on site (Levinson and Zhang 2006). The successful experiments done on ramp metering led to its spreading around the United States. After that, ramp metering evolved into using pre-timed signal controllers. Advanced technology played a major part in providing real-time data, which led to the development of traffic responsive algorithms.

There are a variety of ramp metering algorithms ranging from feedforward algorithms such as Percent-Occupancy (OCC), feedback algorithms such as ALINEA, and system-wide algorithms such as System Wide Adaptive Ramp Metering (SWARM).

1.1. Traffic Flow Theory

Traffic flow studies started in the 1920s in an attempt to discover a way to analyze traffic based on measureable parameters (Gazis and Edie 1968). The relationship between traffic flow, traffic density, and speed is based on the fundamental equation:

$$q = ku \quad \text{Equation 1}$$

Where:

q = Traffic stream flow in veh/hr

k = Traffic stream density in veh/mi

u = Space mean speed in mi/hr

The fundamental diagrams, which describes the relationship between flow, speed, and density are based on the Greenshields model (Zaidi et al., 2016), as shown in Figure 1:

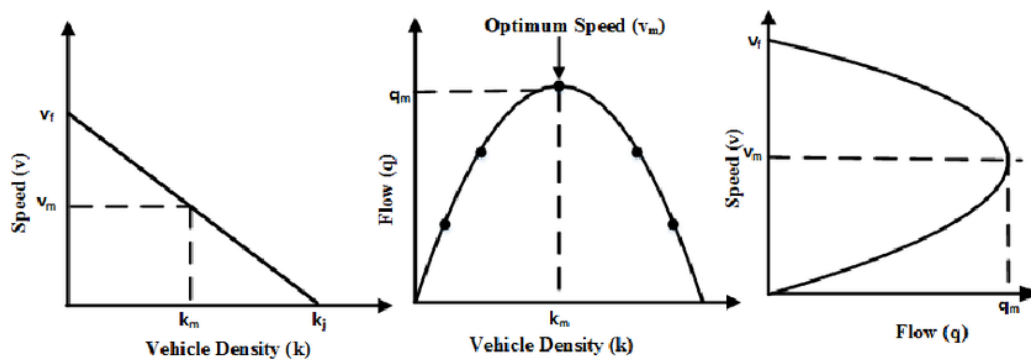


Figure 1 Fundamental diagram of traffic flow (Zaidi et al., 2016)

Greenshields et al., (1993) researched traffic flow, density and speed using photographic measurement methods. They plotted the measured data and postulated a linear relationship between speed and traffic density as shown in Figure 2.

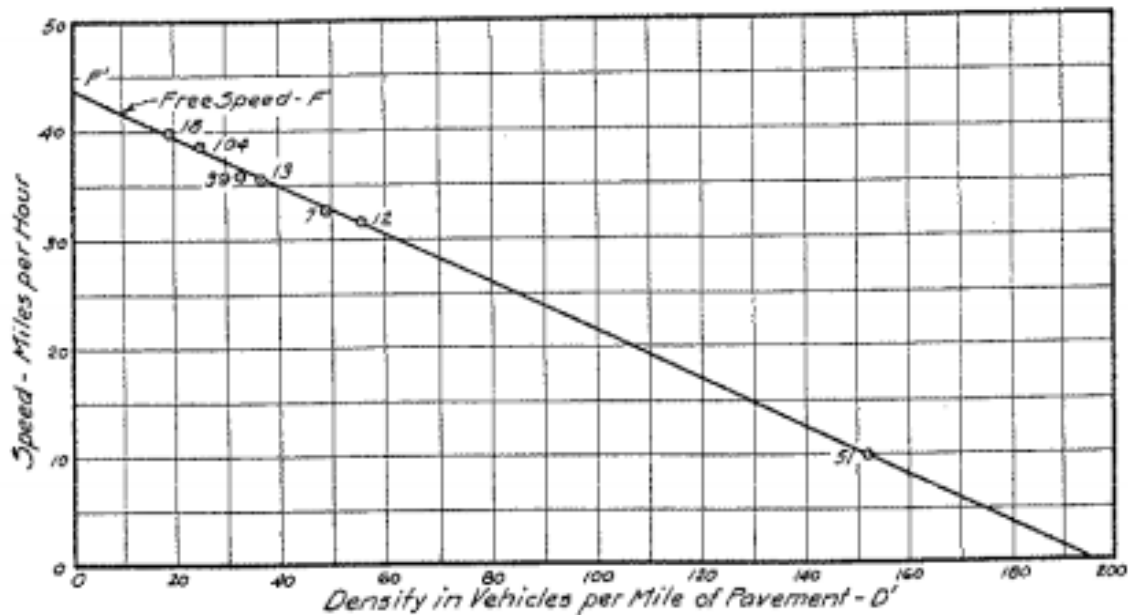


Figure 2 Speed-Density Relationship (Greenshields et al., 1935)

They also computed the relationship between speed and flow. Greenshields used the term “Density-Vehicles per Hour” instead of flow as the term flow was not known at that time. The derived speed flow relationship is parabolic as shown in Figure 3.

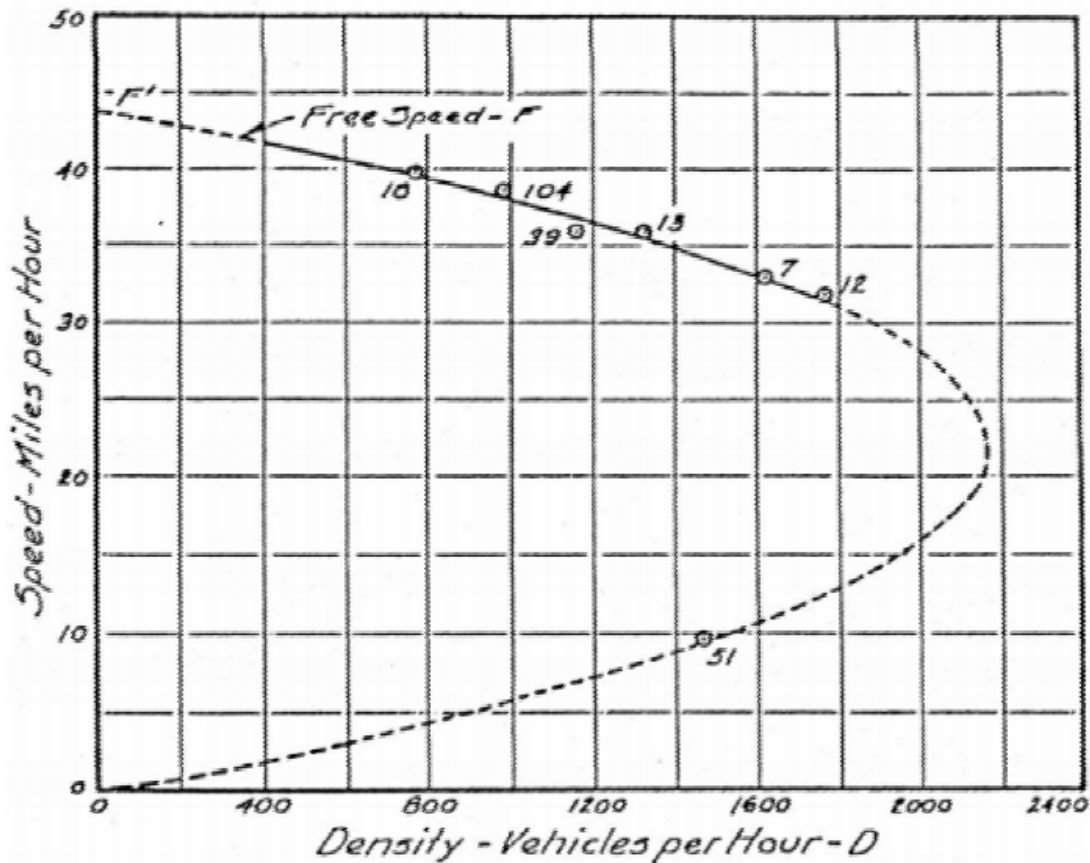


Figure 3 Speed Flow Relationship (Greenshields et al., 1935)

It can be concluded from Greenshields model that the traffic stream has two states: congested and uncongested. The uncongested state is when the traffic density is below the critical density. In this state traffic is stable, more vehicles are served and maximum flow (i.e., capacity) can be achieved. The congested state is when the traffic density exceeds the critical density. In this state the traffic flow is unstable, less vehicles are served and the maximum flow cannot be achieved. Ramp meters function as controllers that try to keep the traffic in uncongested conditions and close to capacity.

1.2. Ramp Metering Strategies

Ramp metering strategies are categorized as traffic responsive or pre-timed. Traffic responsive systems use real-time measurements, whereas historical data are used to determine the

ramp metering rates at pre-timed control (Zhang et al., 2001). Pre-timed control is effective when traffic conditions and congestion are recurrent, but performs poorly when traffic is unpredictable or non-recurrent (Zhang et al., 2001). On the other hand, traffic responsive controllers compute ramp metering rates use real time measurements, and can be more effective during non-recurrent events (e.g., incidents) (Zhang et al., 2001).

There are three types of ramp metering control systems: (i) localized or isolated systems, where each ramp is managed independently without considering other ramps; (ii) system-wide or coordinated systems, in which the traffic conditions of a facility with multiple controlled on-ramps is used to coordinate the metering rate of the on-ramps; (iii) integrated systems, which use other control measures in addition to ramp metering, such as route guidance through variable message signs and signal timings. Integrated systems are very sophisticated and difficult to implement in the field, and no successful implementation has been achieved yet (Zhang et al., 2001).

1.3. Thesis Objectives

The objective of this research is to assess the effect of specific ramp metering algorithms on freeway operations using VISSIM microsimulation software. Through simulation, the effect of each algorithm on specific performance measures were evaluated. Then, the results of each algorithm were compared to each other to find the most suitable algorithm for this freeway.

2. LITERATURE REVIEW

Ramp metering is the operation of controlling the entering traffic into a freeway by using traffic signals on the ramps. Its objective is to regulate the traffic entering the mainline from the on-ramp to prevent the formation of congestion (Jacobson et al., 2006). Also, ramp metering plays a great role in enhancing the performance of the roadway, if implemented correctly. It was initiated as a pre-timed signal controller and in years it evolved to operate as a traffic-responsive signal controller using real-time traffic measurements. There are large varieties of ramp metering algorithms ranging from localized (regulate a single on-ramp) to system-wide or coordinated (regulate a facility with multiple on-ramps) (Jacobson et al., 2006).

2.1. Local Ramp Metering Strategies

Local ramp metering strategies regulate a single on-ramp as an independent system. Early traffic-responsive algorithms were based on feedforward philosophy like Demand Capacity (DC) and Percent-Occupancy (OCC) algorithms. Recent algorithms started using the popular feedback philosophy like ALINEA (Papamichail et al., 2010). The main difference between feedforward (open-loop) and feedback (closed-loop) is that the output of the system is not used in the next iteration in feedforward systems but, it is used in feedback systems. At feedback systems the detectors are usually installed downstream of the on-ramp where the merge occurs (Papamichail et al., 2010).

2.1.1. Demand Capacity (DC)

Demand-Capacity ramp metering strategy also known as DC is a feedforward disturbance compensation strategy which uses flow upstream of the ramp (Masher et al., 1975), (Koble and Samant 1980). This algorithm is used to prevent recurrent congestion (Koble and Samant 1980) and it was field tested in Boulevard Périphérique in Paris (Papageorgiou et al., 1997). DC pre-

defines maximum and minimum ramp metering rates. The maximum $r(k)$ value is set as ramp's cycle length, for example, for cycle length of 4 sec/veh, maximum $r(k) = 900\text{veh/hr}$. The DC is described in Equations 2 and 3.

$$r_{cal}(k) = q_{cap} - q_{in}(k - 1) \quad \text{Equation 2}$$

$$r(k) = \begin{cases} r_{max} & \text{if } r_{cal}(k) \geq r_{max} \\ r_{min} & \text{if } r_{cal}(k) \leq r_{min} \\ r_{cal}(k) & \text{otherwise} \end{cases} \quad \text{Equation 3}$$

$k = 1, 2, \dots = \text{discrete time index.}$

$r_{cal}(k) =$ Calculated admissible ramp flow (veh/h).

$q_{in}(k - 1) =$ Previous measured upstream freeway flow (veh/h) in all lanes.

$q_{cap} =$ Downstream freeway capacity (veh/h).

$r(k) =$ Ramp flow in vehicle per hour (veh/h) during period k .

$r_{min} =$ Minimum allowed ramp flow (veh/h).

From Equation (2) it can be concluded that the ramp flow entering the freeway is computed to achieve downstream capacity. Ramp flow is the difference between the downstream capacity and the upstream flow, therefore, the ramp flow increases (or decreases) with the decrease (or increase) of the upstream flow. The DC algorithm also uses occupancy in determining the freeway congestion state. A critical occupancy value o_{cr} is used to identify if the freeway is congested or not. If the upstream occupancy exceeds the critical occupancy, the ramp flow is reduced to r_{min} .

The downside of this strategy is that it ignores the downstream traffic conditions, as well as variations of capacity due to adverse environmental conditions (e.g. darkness, rain, etc.) (Papamichail et al., 2010).

2.1.2. Percent-Occupancy (OCC)

The Percent-Occupancy ramp metering strategy, also known as OCC, is a feedforward disturbance compensation strategy which uses upstream occupancy of the ramp (Koble and Samant 1980). The OCC assumes that the fundamental diagram between flow and occupancy is linear as shown in Equation 4.

$$q_{in} = \frac{v_f * o_{in}}{g} \quad \text{Equation 4}$$

v_f = Free-flow speed of the freeway (mph).

g = Depends on vehicle length and effective detector length.

o_{in} = Upstream freeway occupancy (percent of time the detector is occupied).

OCC is similar to the DC strategy as it defines maximum and minimum ramp metering rates. Also, a startup occupancy value is set to activate the algorithm, this value is less than o_{cr} and could be set to $2/3 o_{cr}$ (Masher et al., 1975). OCC is interpreted in Equation 5.

$$r(k) = \begin{cases} r_{max} & \text{for } o_{in} \geq \frac{2}{3} o_{cr} \\ q_{cap} - \frac{v_f}{g} * o_{in}(k-1) & \text{for } \frac{2}{3} o_{cr} \leq o_{in} \leq o_{cr} \\ r_{min} & \text{for } o_{in} \geq o_{cr} \end{cases} \quad \text{Equation 5}$$

Since the OCC assumes that the fundamental diagram between flow and occupancy is linear, it is less accurate than the DC strategy (Smaragdis and Papageorgiou 2003). This algorithm was used extensively in Chicago/Minneapolis, and it was field-tested in Boulevard Périphérique in Paris (Papageorgiou et al., 1997), and it has also been evaluated through simulation (Koble and Samant 1980).

2.1.3. Asservissement Linéaire d'Entrée Autoroutière (ALINEA).

Asservissement Linéaire d'Entrée Autoroutière ramp metering strategy, also known as ALINEA, is a feedback-controlled strategy which uses occupancy downstream of the ramp (Papageorgiou et al., 1991). This strategy uses the outputs of its previous iteration as input for its current. The outputs that are used are previous iteration metering rate $r(k - 1)$ and downstream occupancy $o_{out}(k - 1)$. ALINEA uses Equation 6 for calculating the ramp metering rate.

$$r(k) = r(k - 1) + K_R[\hat{o} - o_{out}(k)] \quad \text{Equation 6}$$

Where $r(k)$ is the current ramp metering rate in seconds, $r(k - 1)$ is the previous iteration ramp metering rate in seconds, K_R is a regulator parameter (smoothing factor), \hat{o} is the desired downstream occupancy and o_{out} is the measured occupancy in vehicles per mile. ALINEA is one of the most popular and robust local algorithms.

2.1.4. ALINEA Extensions

ALINEA is a very popular algorithm and many versions have been developed to improve its efficiency. Some of ALINEA's extensions are FL-ALINEA, UP-ALINEA, UF-ALINEA, AD-ALINEA, X-ALINEA/Q, and PI-ALINEA (Shaaban et al., 2016). Each extension has its own unique algorithm and is derived either for certain circumstances in the system or as a new method to control the ramp metering.

FL-ALINEA (Smaragdis and Papageorgiou 2003).

ALINEA measurements use occupancy instead of flow because for a single flow value the system can be either congested or uncongested. In addition, flow is not a stable measurement for the system because it varies depending on the environmental conditions (rain, lighting, etc.). However, occupancy is not directly related to the fundamental traffic flow variables (traffic

volume, density, average speed) because of the ambiguity of the g-factor in equation (4). FL-ALINEA was developed for some situations where flow measurements might be useful for some facilities and easier to specify. FL-ALINEA follows Equation 7.

$$r(k) = \begin{cases} r(k-1) + K_F[\hat{q} - q_{out}(k-1)] & \text{if } o_{in}(k-1) \leq o_{cr} \\ r_{min} & \text{else} \end{cases} \quad \text{Equation 7}$$

Where all variables as described previously.

UP-ALINEA (Smaragdis and Papageorgiou 2003).

UP-ALINEA was developed to account for the lack of measurement detectors downstream of the ramp in which upstream traffic measurements can be used to run ALINEA algorithm. UP-ALINEA estimates the downstream occupancy \tilde{o}_{out} based on the ramp and upstream flow. \tilde{o}_{out} can be estimated using Equation 8.

$$\tilde{o}_{out}(k) = o_{in}(k) \left[1 + \frac{q_{ramp}(k)}{q_{in}(k)} \right] \frac{\lambda_{in}}{\lambda_{out}} \quad \text{Equation 8}$$

Where λ_{in} and λ_{out} represent the number of mainline lanes upstream and downstream respectively, and all other variables as described previously. Then \tilde{o}_{out} (estimated occupancy) is used in the ALINEA equation instead of o_{out} . The UP-ALINEA follows Equation 9.

$$r(k) = r(k-1) + K_R[\hat{o} - \tilde{o}_{out}(k-1)] \quad \text{Equation 9}$$

UF-ALINEA (Smaragdis and Papageorgiou 2003).

UF-ALINEA is similar to DC and FL-ALINEA. It is similar to DC in the sense that it uses upstream flow in controlling ramp flow, except it uses $\tilde{o}_{out}(k-1)$ instead of $o_{in}(k-1)$. It is similar to FL-ALINEA in terms of the equations used; although it estimates the downstream flow \tilde{q}_{out} instead of q_{out} .

$$\tilde{q}_{out} = q_{in} + q_{ramp} \quad \text{Equation 10}$$

$$r(k) = \begin{cases} r(k-1) + K_F[\hat{q} - \tilde{q}_{out}(k-1)] & \text{if } \tilde{o}_{out}(k-1) \leq o_{cr} \\ r_{min} & \text{else} \end{cases} \quad \text{Equation 11}$$

Where all equations as described previously. Although UF-ALINEA is similar to DC, it reacts faster than DC with the use of $\tilde{o}_{out}(k-1)$ instead of $o_{in}(k-1)$.

X-ALINEA/Q (Smaragdis and Papageorgiou 2003).

ALINEA's drawback is that it limits the entering vehicles to the freeway through the on-ramp, which means if the on-ramp exhibits excessive queues it will affect the adjacent streets traffic. X-ALINEA/Q addresses this problem using an override system. Two ramp metering rates are defined in X-ALINEA/Q: ramp flow $r(k)$ for any ALINEA algorithm, and ramp flow calculated from queue length measurements $\dot{r}(k)$. $R(k) = \max[r(k), \dot{r}(k)]$.

$$\dot{r}(k) = -\frac{1}{T}[\hat{w} - w(k)] + d(k-1) \quad \text{Equation 12}$$

Where T is the time interval in seconds, $w(k)$ is the current queue length in vehicles and $d(k-1)$ is the ramp demand of the previous interval in vehicles in the time interval. It is assumed that $d(k) \approx d(k-1)$. The applied ramp metering rate is $R(k) = \max[r(k), \dot{r}(k)]$. By choosing the maximum value between $r(k)$ and $\dot{r}(k)$ the ramp metering will function in a way that optimizes the freeway until the queue length on the on-ramp starts interfering with the adjacent street. The metering rate $r(k)$ controls the ramp flow to reach its capacity. When $\dot{r}(k)$ is greater than $r(k)$ the applied metering rate, $\dot{r}(k)$, will exceed the metering rate that optimizes capacity, $r(k)$, and this will lead the freeway flow to exceed capacity and experience congestion.

PI-ALINEA (Wang, et al., 2014).

PI-ALINEA is a proportional integral controller that resolves a deficiency in ALINEA when a distant downstream bottleneck activates. The proportional and integral terms are shown in Equation 13.

$$u(t) = u_{bias} + \underbrace{K_p e(t)}_{\text{Proportional Term}} + \underbrace{K_I \int e(t) dt}_{\text{Integral Term}} \quad \text{Equation 13}$$

Where $e(t)$ is the error, K_p and K_I are the proportional and integral controller gain parameters. The integral function fixes the error in a small manner and it uses the cumulative error in its calculation. If the error persists for a long period of time, the error builds up and results in oscillatory adjustments if the integral gain controller K_I was high.

ALINEA manages bottlenecks that are located around 1500ft downstream of the ramp. However, if the bottleneck occurs for example one mile downstream of the ramp, ALINEA will not be able to control it efficiently. Proportional Integral systems are better than Integral systems in dealing with slow dynamics (distant bottlenecks). The PI-ALINEA formula is as follows.

$$r(k) = r(k-1) - K_P[o_{out}(k) - o_{out}(k-1)] + K_R[\hat{o} - o_{out}(k)] \quad \text{Equation 14}$$

Where K_P is a regulator parameter and the remaining parameters are defined earlier. PI-ALINEA has been compared to ALINEA using macro-simulation programs in the occurrence of distant bottlenecks. PI-ALINEA achieved optimal traffic conditions and occupancy remained stable, while ALINEA achieved optimal traffic conditions but the occupancy was oscillating. PI-ALINEA has been tested through simulation but not implemented in the field. Gain controllers K_P and K_R with values of 100 km*lane/h and 4 km*lane/h yield good results.

2.1.5. Mixed Control

Ozbay et al., (2004) developed the Mixed Control algorithm in an attempt to manage the ramp flow with the existence of excessive ramp queues. The algorithm was designed to overcome oscillations in the metering rate that were produced using the X-ALINEA/Q algorithm. Ozbay et al., (2004) designed an algorithm which takes queue length and freeway demand into account at the same time. The following equations are the proposed mixed control algorithm.

$$u = G^{-1}[-F - Ke(k)] \quad \text{Equation 15}$$

$$F = \text{sign}[\rho(k) - \rho_{cr}]w_1 \left\{ \rho(k) - \rho_{cr} + \frac{T}{\Delta x} [-q_{out}(k) + f_1(k)] \right\} \\ + w_2 [queue_{ramp}(k) + Tf_2(k)] \quad \text{Equation 16}$$

$$G = \left\{ \text{sign}[\rho(k) - \rho_{cr}]w_1 \frac{1}{\Delta x} - w_2 \right\} T \quad \text{Equation 17}$$

Where:

$f_1(k)$	The flow entering the freeway section at time step k
$f_2(k)$	The flow entering the ramp at time step k
$u(k)$	Metered ramp flow at time step k
$\rho(k)$	Freeway density for upstream plus downstream section
ρ_{cr}	The critical value of section density (veh/mile)
$q_{out}(k)$	The flow leaving the freeway section at time step k
$queue_{ramp}(k)$	Queue length on the ramp at time step k
w_1, w_2	Weight factors, $w_1 - w_2 = 1$
K	Control gain, $0 < K < 1$

T	Time step duration
Δx	Length of the freeway section

2.2. Coordinated Ramp Metering Strategies

Coordinated ramp metering algorithms have been used to manage freeway facilities that experience congestion. Coordinated ramp metering algorithms such as FUZZY, HERO, METALINE, etc. coordinate multiple metered ramps to prevent and dissolve the occurrence of congestion. They assign metering rates to each metered ramp while considering the benefit of the whole facility and not only its respective local segments. The following sections describe several coordinated ramp metering algorithms that have been tested and used in the field.

2.2.1. METALINE

METALINE is the integral coordinated system version of ALINEA (Papageorgiou et al., 1990). It turns the ALINEA equation into a vector.

$$r(k) = r(k-1) - K_{LQI}^1[\rho(k) - \rho(k-1)] - K_{LQI}^2[\hat{\rho}(k) - \hat{\rho}_d] \quad \text{Equation 18}$$

Where ρ is the vector of densities and K_{LQI}^1 and K_{LQI}^2 are gain matrices. The gain matrices values are based on the desired traffic performance. METALINE was implemented in Boulevard Périphérique in Paris, France as an incident controlling algorithm. METALINE and ALINEA were evaluated using macroscopic simulation program called META. In the presence of unexpected incidents, METALINE was superior in dissolving congestion faster. When both were compared in normal conditions (recurrent congestion) they had approximately the same performance.

2.2.2. Zone Algorithm

The Zone algorithm is based on balancing the entering and exiting flows in a pre-defined freeway section called metering zone (Stephanedes 1994, Zhang et al., 2001). The freeway is divided into 3 to 6 miles-long metering zones and each metering zone may have metered and non-metered ramps and off-ramps. The zone algorithm uses equation 15 to control and balance the metering zone.

$$A + U + M + F = X + B + S \quad \text{Equation 19}$$

Where:

- A is the upstream mainline volume (veh/hr) (measured).
- U is the total volume from non-metered ramps (veh/hr) (measured)
- M is the total volume from metered ramps (veh/hr) (calculated)
- F is the total metered freeway to freeway volumes (veh/hr) (calculated)
- X is the sum of off-ramp volumes (veh/hr) (measured)
- B is the downstream bottleneck capacity (veh/hr)
- S is the space available within the zone (veh/hr)

By setting $S = 0$ the maximum metering rate ($M + F$) is shown in equation 16 and each ramp metering rate is weighted from the maximum metering rate (shown in Equations 16 and 17);

$$M + F = (X + B) - (A + U) \quad \text{Equation 20}$$

$$R_r = f_r(M + F) \quad \text{Equation 21}$$

Where R_r is the ramp metering rate, and f_r is the ramp share factor from the maximum metering rate

2.2.3. Fuzzy Logic Control (FLC)

Fuzzy Logic was originally developed by Zadeh in 1965 as a way to process industry and produce home appliances. It was also implemented in automobiles and construction business (Zadeh 1965). Fuzzy Logic Control (FLC) is a simple algorithm based on adjusting weighted controlling traffic parameters to optimize the system operation (Taylor et al., 1998). It has been field tested on the A12 freeway between The Hague and Utrecht. FLC showed an increase in travel speed and bottleneck capacity by 35% and 5-6% respectively. One of the benefits of FLC is that it is less sensitive to imprecise or missing inputs since it uses qualitative inputs instead of quantitative inputs resulted from a process called “fuzzification”.

The FLC process comprises of the following three steps:

1. Fuzzification: preprocesses and transforms quantitative inputs into qualitative data.
2. Rule Evaluation: fuzzification results are analyzed through rules similar to human reasoning in which the most weighted rule is implemented.
3. Defuzzification: turns rule evaluation results into numerical inputs that are implemented as ramp metering rates.

Fuzzification:

The variables that are used in the fuzzification process are based on 20-second intervals measured in the previous period, except for the ramp queues, which use multiple previous 20-seconds time intervals. These measures are translated into inputs to one of five classes that describe the metering rate outcome [Very Small (VS), Small (S), Medium (M), Big (B), and Very Big (VB)]. After that the membership degree for these inputs is evaluated to its class. Membership degree is the amount of how much each class is correct on a scale from zero to one. In measuring

the membership degree, two scaling parameters are assigned as the Low Limit (LL) and High Limit (HL) for each variable. These limits are used to calculate the scaled crisp variable, x , as shown in Equation 22.

$$\text{Scaled Crisp Variable} = x = \frac{\text{crisp variable}}{HL-LL} - \frac{LL}{HL-LL} \quad \text{Equation 22}$$

Each of the five fuzzy classes is indicated with a function $f_i(x)$, where the centroid is C_i , β_i is the base width, and i indicates the class. Classes S, M, and B are defined by a triangle with two sides and a base of $2\beta_i$. The triangle side slope is $\pm 1/\beta_i$ and is centered at C_i . The membership degree for variables in classes S, M, and B are calculated using Equation 23.

$$f_i(x) = \begin{cases} \frac{1}{\beta_i}(x - C_i + \beta_i) & \text{for } C_i - \beta_i < x < C_i \\ -\frac{1}{\beta_i}(x - C_i - \beta_i) & \text{for } C_i < x < C_i + \beta_i \end{cases} \quad \text{Equation 23}$$

The VS and VB classes are defined by a right triangle with base of β_i and can be calculated using Equations 24 and 25.

For VS:

$$f_i(x) = \begin{cases} 1 & \text{for } x < 0 \\ -\frac{1}{\beta_i}(x - \beta_i) & \text{for } 0 < x < \beta_i \end{cases} \quad \text{Equation 24}$$

For VB:

$$f_i(x) = \begin{cases} \frac{1}{\beta_i}(x - 1 + \beta_i) & \text{for } 1 - \beta_i < x < 1 \\ 1 & \text{for } x > 1 \end{cases} \quad \text{Equation 25}$$

For example, Figure 4 shows the membership degree for occupancy where LL= 8 and HL = 18.

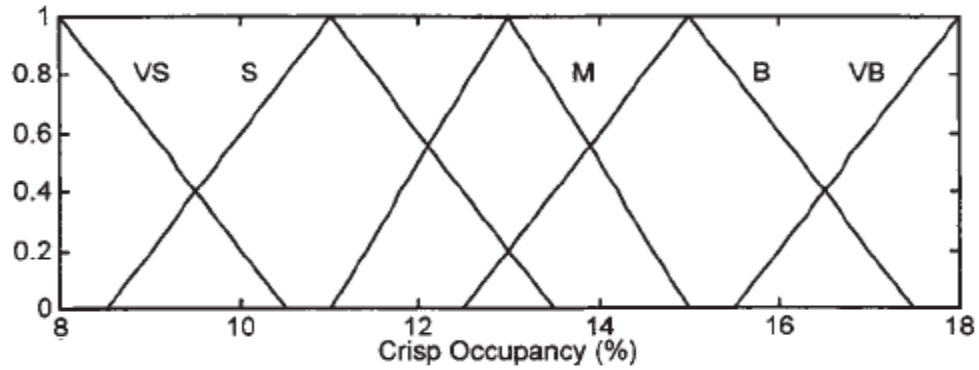


Figure 4: Occupancy membership degree for each class (Taylor et al., 1998)

Table 1 Fuzzy Logic Input Variables (Taylor et al., 1998)

Variable	Description
OC	Main line occupancy just before ramp outlet
DO	Downstream occupancy of nearest bottleneck prone section
UO	Occupancy from adjacent upstream station
SP	Speed for main line just before ramp outlet
DS	Downstream speed of nearest bottleneck prone section
QO	Ramp queue occupancy smoothed over past 6 samples
AQO	Advanced queue detector occupancy over past sample

Table 1 presents the parameters used to define the rules that govern the FLC algorithm built by Taylor et al., (1998) for their field test. Each parameter is assigned a membership function and then used into a rule solely or alongside another parameter.

Rule Evaluation:

In this step, the fuzzified input variables are tested with specific rules. Each rule has a weight reflecting its importance. There are 17 rules that were considered important in simulation testing shown in Table 2.

Table 2 FUZZY Logic Controller rules (Taylor et al., 1998)

Rule	Default Rule Weight	Premise	Metering Rate Outcome
1	1.0	OC_VB	VS
2	1.0	OC_B	S
3	1.0	OC_M	M
4	1.0	OC_S	B
5	1.0	OC_VS	VB
6	2.0	SP_VS, OC_VB	VS
7	1.0	SP_S	S
8	1.0	SP_B	B
9	2.0	SP_VB, OC_VS	VB
10	1.0	UO_M	M
11	2.0	UO_S	B
12	2.0	UO_VS	VB
13	2.5	DS_VS, DO_VB	VS
14	2.5	DS__S, DO_B	S
15	1.0	DS_M, DO_M	M
16	2.0	QO_VB	VB
17	3.0	AQO_VB	VB

Rules from 1 to 5 are the baseline of metering rates. The minimum weight for these must be at least 0.1, however, other rules could be 0. Rules 6 to 9 reinforce Rules 1 to 5 as they use both speed and occupancy measurements to find the congestion index and adjust metering rates. The comma denoted between occupancy and speed means the logic AND; also, the minimum of these is used. Rules 10 to 15 take into account that traffic waves move forward in uncongested conditions and under congested conditions traffic waves travel backwards. Rules 16 and 17 take into account the development of queues on the ramp.

Defuzzification:

The goal of defuzzification is to convert the rules results into ramp metering rates. The metering rates resulting from the defuzzification follow Equation 26.

$$\text{Metering Rate} = \frac{\sum_{i=1}^N w_i c_i I_i}{\sum_{i=1}^N w_i I_i} \quad \text{Equation 26}$$

Where:

w_i = Weight of the i-th rule

c_i = The centroid of the output class

I_i = The implicated area of the output class

2.2.4. HEuristic Ramp-metering CoOrdination (HERO)

HEuristic Ramp-metering CoOrdination, also known as HERO, is a linked algorithm that uses master-slave structure to manage on-ramp metering rates (Papamichail and Papageorgiou 2008). HERO assigns the Master role to the downstream on-ramp where the bottleneck occurs. This bottleneck occurs because ALINEA implements queue control when there is insufficient ramp storage. HERO assigns upstream on-ramps as slaves and uses their ramp storage for the master ramp. HERO coordinates and controls the upstream on-ramp (Slave) metering rate by assigning minimum queue length w_{\min} . The upstream on-ramp metering rates are calculated based on Equations 27 and 28.

$$q_o^{LC}(k_C) = -K_w[w_{\min,o} - w_o(k_C)] + d_o(k_C - 1) \quad \text{Equation 27}$$

$$q_o(k_C) = \max\{\min\{q_o^r(k_C), q_o^{LC}(k_C)\}, q_o^w(k_C)\} \quad \text{Equation 28}$$

Where $q_o(k_C) = r(k)$, K_w is a control parameter set as $1/T_c$ or less for smoother control action and T_c is the control sample time. When HERO is activated, both upstream and downstream ramp rates are regulated so that the upstream and downstream relative queue lengths stay close to

each other. The minimum queue length which is assigned to the upstream on-ramp is updated ever T_c until the Master ramp relative queue falls below the activation threshold.

2.2.5. System Wide Adaptive Ramp Metering (SWARM)

The SWARM algorithm comprises of two independent algorithms where the more restrictive of the two is used to control the ramp metering rate. The first algorithm is called SWARM1 and it is based on forecasting and system-wide apportioning (Paesani et al., 1997, Bogenberger and May 1999). It uses traffic density to maintain the roadway under the saturation density level. It forecasts the density trends from previous interval data by using linear regression and Kalman filtering. Then it computes the excess density in the freeway and uses it to calculate the volume reduction value or volume excess, which are then distributed to the upstream ramp meters as metering rates, by using weight factors based on ramp demands and queue storage. SWARM1 follows equations 29 and 30.

$$\text{Target Density} = (\text{Current Density}) - \left(1/T_{\text{Crit}}\right) * (\text{Excess Density}) \quad \text{Equation 29}$$

$$\text{Volume Reduction} = (\text{Local Density} - \text{Target Density}) * (\# \text{ of Lanes}) * \\ (\text{Distance to next Station}) \quad \text{Equation 30}$$

Where T_{Crit} is the time into the future in which the density is forecasted.

The second algorithm is called SWARM2, and it is a local traffic responsive system. It turns measured densities into metering rates using linear conversion. However, SWARM2 is dependent on the accuracy of the density predictions in SWARM1 to operate effectively.

2.3. Ramp Metering Field Evaluations

This section presents real life field results of implementing ramp metering algorithms on freeways. Field evaluations provide the effects of ramp metering using real life driving behavior. Also, the algorithms can be assessed when there are unexpected incidents such as crashes, bad weather, detector malfunctions and unexpected driver behavior.

2.3.1. Monash Freeway, M1 and M3 freeways in Queensland Australia (HERO)

Papamichail et al., (2010) evaluated HERO algorithm that was implemented by VicRoads on Monash freeway in 2008. The evaluation was done between the previous ramp metering strategy (not specified) and HERO on six consecutive on-ramps. The implementation of HERO showed an increase in the morning average flow rate by 4.7% and average speed by 35%; while during the afternoon peak, the flow rate and speed increased by 8.4% and 58.6% respectively. The results also showed improvement in travel time reliability and mean speed deviation. Faulkner et al., (2014) evaluated the implementation of HERO algorithm along M1 and M3 freeways in Queensland, Australia. M1 and M3 ramp meters were upgraded from fixed-time control to HERO control. The analyzed section contained five consecutive metered on-ramps leading to Brisbane central business district (CBD). After implementing HERO, the morning peak travel speed increased by 7%, the throughput increased by 4%, and the travel time reliability increased from 19% to 56% excluding traffic incidents compared to the fixed-time strategy. Figure 5 presents the occupancy heat plots of before (Figure 5a) and after implementing HERO (Figure 5b).

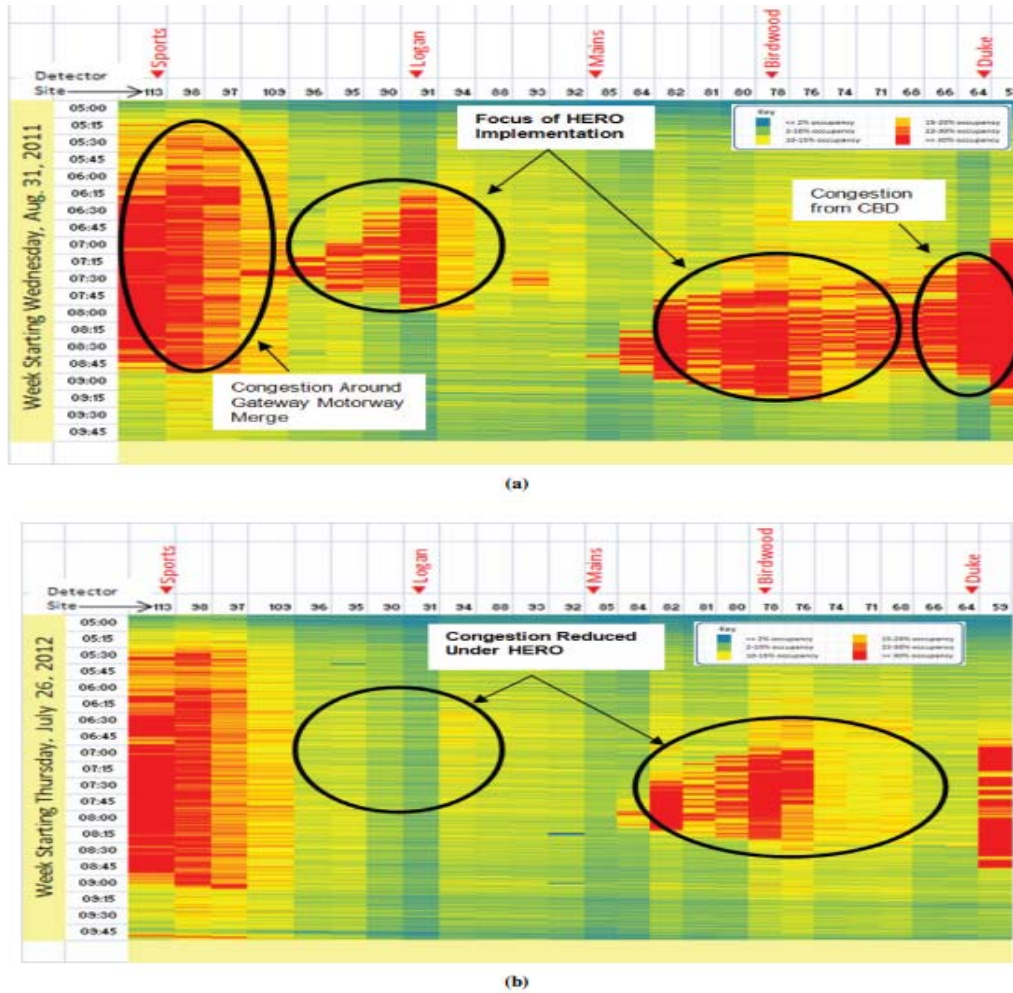


Figure 5: Occupancy heat plots (a) before and (b) after implementing HERO

(Faulkner et al., 2014)

2.3.2. A6W Motorway, Paris – ALINEA

The upgrade of A6W Motorway in Paris was under the scope of EURAMP2006 project. ALINEA was tested on a corridor in this motorway, which included five on-ramps and its length is approximately 20km (Bhourri et al., 2013). The test took place between September 2005 and January 2007. Days with detector failures, weekends, holidays, and accidents were discarded from the analysis. The remaining dataset included 11 and 10 days for NO Control and ALINEA respectively. The implementation of ALINEA showed a reduction in total time spent by 9.8% and an increase in mean speed by 4.3%. In terms of travel time reliability, ALINEA improved the

motorway by 31% in its Misery Index (MI), 37% in Buffer Index (BI), and 28% in Planning Time Index (PTI).

2.3.3. I-90 and I-405, Seattle – Fuzzy Logic control

The Fuzzy Logic Control (FLC) was compared to a local algorithm and the Bottleneck Algorithm on I-90 and I-405 respectively (Taylor and Meldrum 2000). I-90 exhibits moderate congestion and I-405 exhibits heavy congestion. On I-90, FLC decreased occupancy by 8.2%, and increased throughput by 4.9% compared to the local algorithm. However, longer ramp queues were observed in the fuzzy algorithm compared to the local algorithm. On I-405, FLC resulted in slightly increased occupancy and throughput compared to the Bottleneck Algorithm. However, ramp queues were much longer when implementing the Bottleneck algorithm compared to the FLC.

2.3.4. Assessment of FUZZY LOGIC on A12 motorway in Holland

Taale et al., (1996) evaluated ALINEA, Fuzzy Logic, and an algorithm named Rijkswaterstaat (RWS), on A12 motorway from The Hague to Utrecht in Holland. The evaluation results showed the following capacities: ALINEA: 4,000 veh/hr, RWS: 4,048 veh/hr, and Fuzzy: 4,256 veh/hr. The travel times were: ALINEA 6.2 minutes, RWS 6.0 minutes, and Fuzzy 3.9 minutes.

2.4. Simulation Evaluations

Simulation is a great tool to assess any strategy before implementing them in the field. Simulation is feasible in terms of money and time compared to field evaluations and provide an environment where outside inferences can be eliminated such as weather, incidents, and change in traffic patterns.

2.4.1. Comparison of HERO, ALINEA, and PI-ALINEA

HERO, ALINEA and PI-ALINEA with queue control were compared by using a macroscopic simulation software named METANET (Papamichail and Papageorgiou 2008). The test consisted of two on-ramps. Multiple scenarios were tested and the results with queue control are shown in Table 3.

Table 3 ALINEA, PI-ALINEA, and LC (HERO) simulation results

Strategy	Max Queue (veh)	TWT (veh.h)	TTS (veh*h)	TTS' (veh*h)	%decrease in TTS'
No Control	-	0	983	817	0
ALINEA	50	78	908	742	9.2
PI-ALINEA	50	67	915	749	8.3
HERO (30%-15%)	50	98	874	708	13.3
HERO (80%-40%)					

Where TWT is the total waiting time vehicles spent on the on-ramp waiting to be served, TTS is the total time spent by each vehicle in the facility, and TTS' is the total time spent by each vehicle in the facility disregarding the first 30 minutes for this evaluation.

The HERO percentages shown in Table 3 represent activation threshold and deactivation threshold respectively the first being activation threshold. Table 3 shows that HERO algorithm yielded a lower TTS and a higher TWT compared to ALINEA and PI-ALINEA. In addition, it was found that the change in activation and deactivation threshold of HERO did not yield significant change in traffic conditions.

2.4.2. Comparison of ALINEA, ZONE, and Bottleneck

ALINEA, ZONE, and Bottleneck algorithms were evaluated on I-405 freeway segment in California using PARAMICS microsimulation model (Chu et al., 2004). All of the algorithms were

superior to the currently deployed fixed-time control. The simulation results of the algorithms compared to fixed time results are shown in Table 4.

Table 4 ALINEA, ZONE, and BOTTLENECK simulation results

Conditions and Algorithms	Vehicles-hours Traveled Change	Average Mainline Travel Time Change	Total On-ramp Delay Change
Heavily Congested			
ALINEA	-4.8%	-5.1%	+24.9%
ZONE	-4.3%	-4.2%	+43.5%
BOTTLENECK	-5.2%	-6.6%	+51.9%
Less Congested			
ALINEA	-3.0%	-3.1%	+10.83%
ZONE	-0.2%	-1.1%	+77.5%
BOTTLENECK	-1.5%	-2.6%	+53.8%
Severe Incident			
ALINEA	-1.9%	-2.3%	+20.3%
ZONE	+0.5%	+0.4%	+58.1%
BOTTLENECK	-0.4%	-0.5%	+34.2%
Less Sever Incident			
ALINEA	-1.1%	-1.4%	+27.5%
ZONE	-1.4%	-2.2%	+58.6%
BOTTLENECK	-1.3%	-1.6%	+30.4%

Table 4 shows that during a heavy congested scenario BOTTLENECK yielded lower vehicle-hour travel times and average mainline travel time than ALINEA and ZONE. However, the BOTTLENECK algorithm resulted in higher on-ramp delay than ZONE and ALINEA. During less congested conditions, ALINEA yielded lower vehicle-hour travel times and average mainline travel time than BOTTLENECK and ZONE. In addition, ALINEA had lower on-ramp delay than

BOTTLENECK and ZONE. During severe incidents the algorithms had the same rankings as in less congested conditions, however, in ZONE travel times increased compared to the No Control scenario. During less severe incidents ZONE yielded the lower travel times than BOTTLENECK and ALINEA. However, ZONE had the highest on-ramp delay than BOTTLENECK and ALINEA.

2.4.3. Twin Cities, Minnesota Coordinated Ramp Metering Assessment

Kwon et al., (2001) evaluated the implementation of the Incremental group coordination algorithm (Denver Algorithm), the Minnesota algorithm, and the Fuzzy Logic algorithm using a macroscopic model called Kronos. The evaluation was conducted on the I-69 freeway in the metropolitan area of the Twin Cities, Minnesota. The Fuzzy and the Denver algorithms have the queue override policy, which means that if the queue reaches a certain threshold the metering rates become less restrictive. However, the Minnesota algorithm does not have this rule. The fuzzy algorithm produced higher mainline vehicle miles when the demand was distribute in the peak period, but had lower mainline vehicle miles when the demand was concentrated compared the other two algorithms. The Minnesota algorithm resulted in more vehicle throughput when the demand was concentrated in the peak period, but also in higher ramp delays. It was concluded that the Fuzzy logic algorithm was superior to the other two algorithms in balancing the metering rates with the ramp queue length.

2.5. Vissim Software

This thesis used VISSIM for evaluating the ramp metering algorithms along the I-35 corridor. Vissim is a microscopic simulation program that can simulate traffic facilities under various conditions of vehicle demand composition, route decision, and signal control (PTV Group 2016). Vissim can simulate signal control to be responsive to traffic conditions using the Vehicle

Actuated Program (VAP). Vehicles move through the facility using a traffic flow model. Vissim uses a psycho-physical perception model developed by Wiedemann (1974). The concept of the Wiedemann model is that a driver driving faster than the leading vehicle will slow down until reaching the leader, and then drive slower than the leading vehicle. After that, the driver starts to accelerate slightly to match the leading vehicle. The Wiedemann car following model is illustrated in Figure 6.

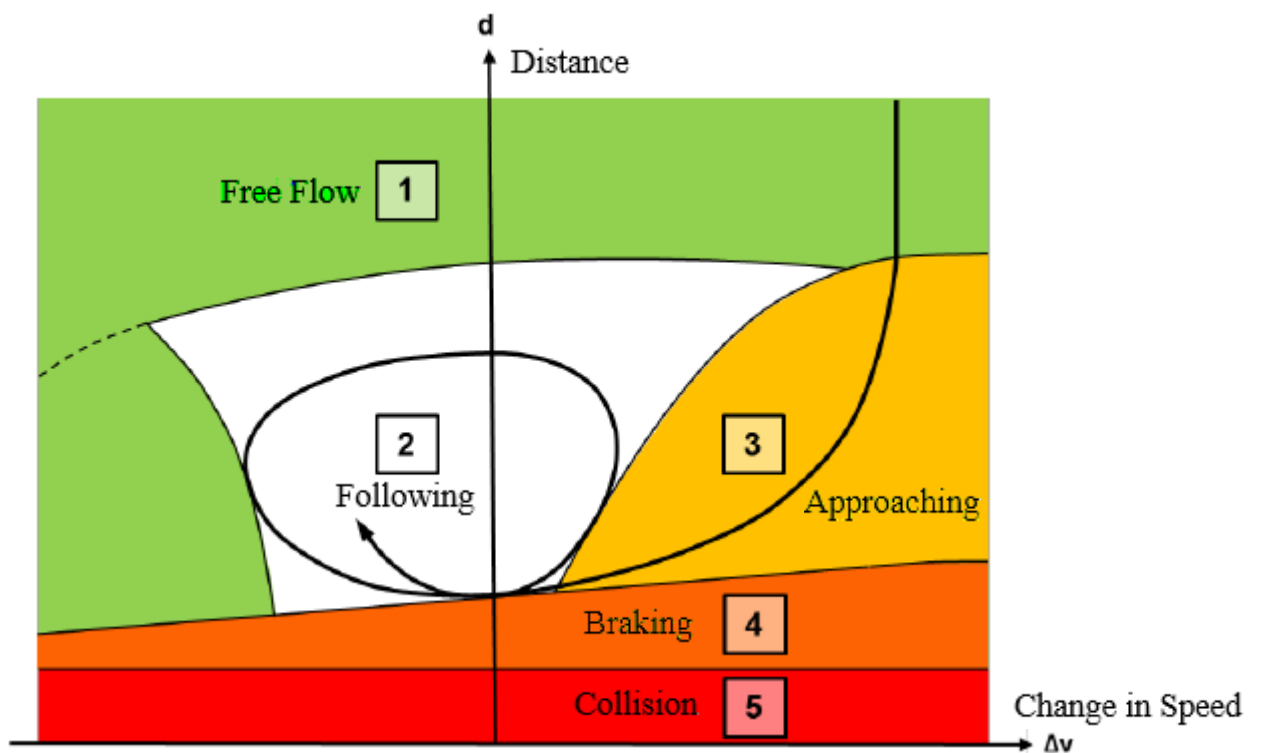


Figure 6 Wiedemann (1974) car following model (PTV Group 2016)

Vissim provides two car following models: Wiedemann 74, and Wiedemann 99. The Wiedemann 74 model is more focused on urban streets and merging locations. The Wiedemann 99 model is focused on freeways and it includes nine parameters that are adjusted to calibrate the model to real life data. A description of the Wiedemann 99 parameters along with their default values is presented in Table 5.

Table 5 Wiedemann 99 car following model parameters

VISSIM Code	Parameters Description	Default Values
CC0	Standstill Distance: Desired distance between two concurrent vehicles at zero speed.	4.92 ft.
CC1	Headway Time: Desired time in seconds between two concurrent vehicles.	0.90 sec
CC2	Following Variation: Distance over safety distance a following vehicle requires before moving closer to the lead vehicle.	13.12 ft.
CC3	Threshold for Entering 'Following' State: Time in seconds before a following vehicle start to decelerate to reach safety distance.	-8.00 sec
CC4	Negative 'Following' Threshold: The negative variation in speed between two concurrent vehicles.	-0.35 ft./s
CC5	Positive 'Following' Threshold: The positive variation in speed between two concurrent vehicles.	0.35 ft./s
CC6	Speed Dependency of Oscillation: Influence of distance on speed oscillation.	11.44 1/(ft.*s)
CC7	Oscillation Acceleration: Oscillation during acceleration	0.82 ft./s ²
CC8	Standstill Acceleration: Desired acceleration starting from standstill	11.48 ft./s ²
CC9	Acceleration with 50mph: Desired acceleration at 50mph	4.92 ft./s ²

2.6. Summary of literature review

Based on the previous literature it can be seen that ramp metering is very effective in maintaining free flow conditions and dissolving congestion if applied correctly on freeways. There are many ramp metering algorithms available in the real world ranging from localized control to coordinated control. Microsimulation modeling helps in assessing the effects of implementing ramp metering on a freeway and evaluate its operational efficiency.

3. METHODOLOGY

This chapter describes the methodology undertaken to complete this research. First, the simulation model development and calibration are presented. Then, the actual simulation process is discussed. After that, the tested ramp metering algorithms are discussed. Finally, the performance measures that were used for the evaluation are explained.

3.1. Model Development and Calibration

The Kansas Department of Transportation (KDOT) added ramp meters on I-35 southbound (at 7th Street Trafficway, Southwest Boulevard, 18th St. Expressway, and 67th Street junctions) in the summer of 2017 (Figure 4). These on-ramps have been identified as locations where merge bottlenecks are created. KDOT is planning to use a ramp metering algorithm which is based on speed variation; however, the algorithm is proprietary and its specifications are not known. The facility geometrics (number of lanes, link lengths, ramps, speed limits, etc.), detector locations and traffic data for a 12-month period were modeled in VISSIM. KDOT provided an initial Vissim file with the geometry and some traffic data, which were used as a basis for this research. The I-35 facility was calibrated to replicate the original conditions of the freeway, which do not include ramp metering. The goal of the calibration was to match the simulated and the real world vehicle speeds at the junctions where ramp metering is installed within a certain percent of accuracy. The calibration was done by using data from a single day from the KC Scout Portal.

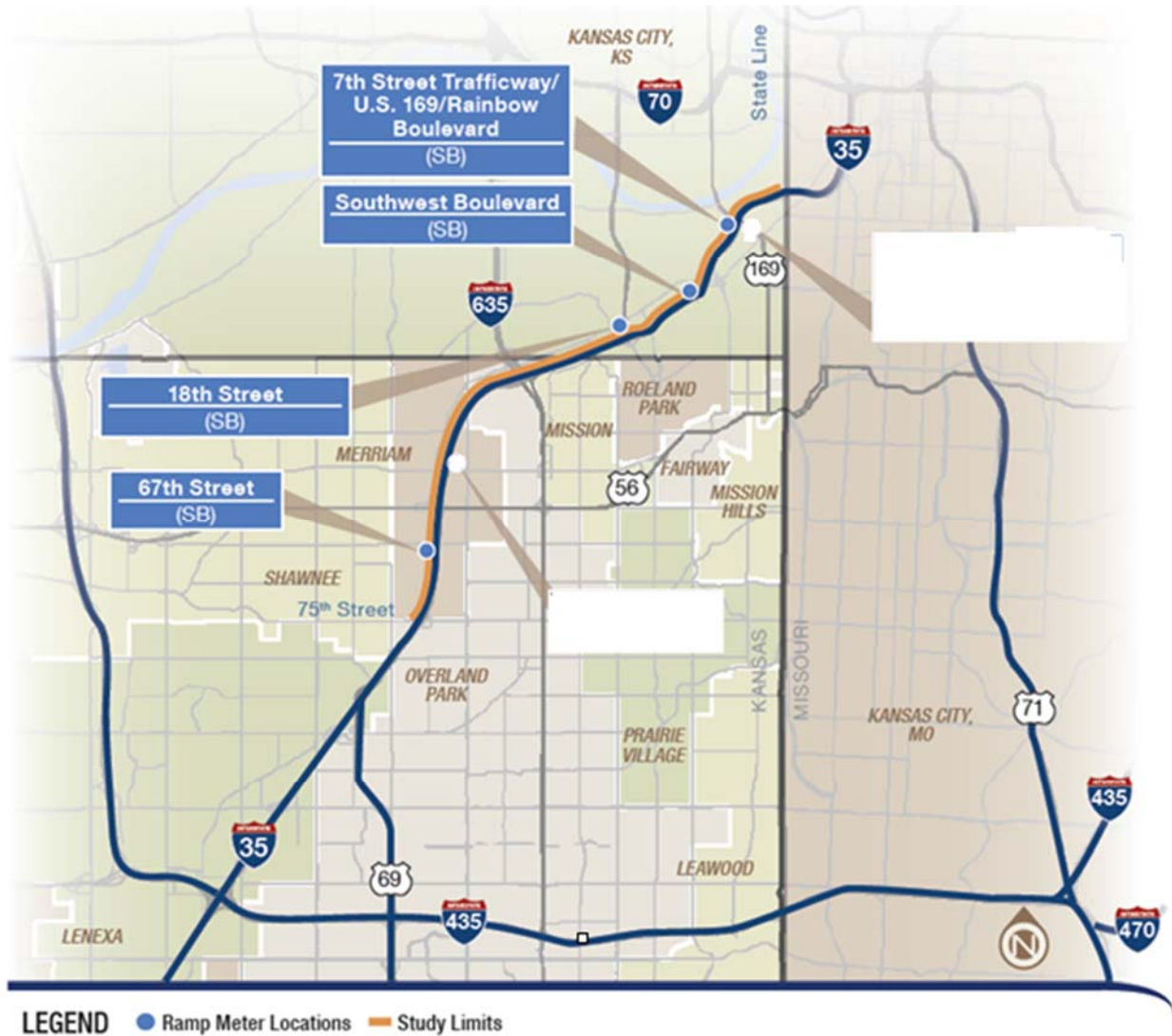


Figure 7 I-35 Map and its Metered Junctions (Provided by KDOT and KCScout)

3.2. Selected Ramp Metering Algorithms

Based on the literature, one system-wide ramp metering algorithm (HERO), and one localized ramp metering algorithm (ALINEA) were selected for evaluation in this thesis. Both of these algorithms have shown efficiency in the field and simulation testing; therefore, it is desired to evaluate their impacts on the I-35 freeway facility. Three control scenarios were simulated and evaluated in VISSIM:

- No Control: This scenario assumes no ramp metering control in the facility.
- HERO Algorithm: In this control scenario the HERO coordinated algorithm was implemented at the three consecutive junctions (7th Street Trafficway, Southwest Boulevard, and 18th Street) on I-35 southbound. In the remaining isolated ramp (67th St), the ALINEA algorithm was implemented.
- ALINEA Algorithm: In this control scenario the ALINEA algorithm was implemented on the all ramps mentioned previously.

3.3. Simulation Process

VISSIM is a microscopic simulation tool that requires each scenario to run multiple times with various random seed numbers in order to replicate real-world fluctuations due to driver behavior. Each control scenario would run 60 demand scenarios in the simulator. The 60 demand scenarios represent 5 weekdays * 12 months. Each demand scenario is modeled as the average weekday demand of its respective month e.g. the average demand of four Mondays in January will represent the Monday demand scenario in January. Days when major detector failures occur or significant incidents happen, such as accidents, roadwork, and snow, were discarded. The traffic data used to develop the demand scenarios were obtained through the KC Scout Portal. Each demand scenario ran four times with different random seed numbers, to account for variations of the same day within the same month (e.g., four Mondays in January).

3.4. Evaluation Results

The VISSIM simulation ran four times with different random seeds for the three control strategies and average performance measures obtained from the software were calculated. The following performance measures were generated from VISSIM to evaluate the control strategies.

- Travel Time (TT): The amount of time a vehicle needs to clear the freeway.
- Travel Time Index (TTI): The ratio between the actual travel times to the travel times in free flowing conditions. It follows Equation 31.

$$TTI = \frac{TT_{Actual}}{TT_{FreeFlow}} \quad \text{Equation 31}$$

- Buffer Time (BT): The extra time needed above the average travel time for the user to arrive on time 95% of the time or at the 95th percentile travel time (TT₉₅).

$$BT = TT_{95} - TT_{Mean} \quad \text{Equation 32}$$

Where (TT_{Mean}) is the mean travel time.

- Buffer Index (BI): The ratio between buffer time (BT) and mean travel time (TT_{Mean}).

$$BI = \frac{BT}{TT_{mean}} \quad \text{Equation 33}$$

- Mean Speed (MS): The average of the vehicles speed in the freeway segment. This parameter is measured at the location where bottlenecks occur.
- Throughput: The rate in which the vehicles are served in the freeway. This parameter is measured at the location where bottlenecks occur.
- Queue Length and Waiting Time (WT): It is the length of the vehicles queue of and the total time spent waiting on the on-ramps to be served.
- Congestion Duration: The amount of time the freeway exhibits congestion until it recovers to free flow conditions.

4. DATA COLLECTION AND SCENARIO DEVELOPMENT

In this chapter the process of data collection and demand scenario development are discussed. Firstly, descriptions of the modeled facility and the data collection are provided. Next, the formatting and screening of the data for incidents and adverse weather conditions are explained. Finally, the data reduction and demand scenario development are presented.

4.1. Facility and Data Overview

The facility analyzed in this thesis is the I-35 southbound corridor from Cambridge Dr. to 75th Street (Figure 7). Traffic volume and speed data at the mainline, on-ramp and off-ramp were obtained from the KC Scout Portal (<http://www.kcscout.net/KcDataPortal>). Weekdays (excluding holidays) were considered for this analysis and the data were obtained from April 1st 2016 to March 31st 2017. The peak period was considered as the analysis period. Based on the detector's speed readings, the peak period was identified to start at 3:15pm and end at 6:15pm. For simulation purposes, the analysis period was defined to be from 2:50pm to 6:35pm.

4.2. Data Formatting and Screening

After collecting the data from the KC Scout Portal, a screening process was performed to ensure that the data acquired corresponded to days with recurring congestion due to excess demand, and not due to traffic incidents or adverse weather.

Days with traffic incidents (minor, major, stalled vehicle, etc.) along the facility were excluded as these cannot be modeled into VISSIM. The data for traffic incidents were obtained from the KC Scout Portal. The portal provides information on the location, start time, and end time of the incident. Days with traffic incidents that occurred within 2 miles downstream of a detector, and 1 hour before or during the peak period were excluded from the analysis. Traffic incident data

were collected from April 1, 2016 to March 31, 2017. Since, the simulation period is from 2:50pm to 6:35pm the incident screening period was considered to be 1:50pm to 6:35pm. The remaining data were retained for further screening due to adverse weather.

Adverse weather conditions were considered to occur during snow, fog, and precipitation of more than 0.20 inches. Weather data were obtained from (www.wunderground.com) for the 12-month period April 1, 2016 to March 31, 2017 in Kansas City using the MCI airport code. Days when the weather was or considered adverse between 2:20pm and 7:05pm were removed from the analysis. The screened data were arranged in an excel spreadsheet based on the month and weekday to be used for the following steps.

4.3. Data Reduction and Demand Scenario Development

The dataset obtained from the screening process was used in developing the demand scenarios. A demand scenario represents the average input volume of one weekday during a particular month in 5-minutes increments. All detector data were divided by day of the week, and month of the year, so for example data were grouped for every Monday in April. Then the volumes for each 5-minute interval were averaged across the same weekdays of the month. The result was the average volume in 5-minute intervals that represent all Mondays in April.

After averaging the volumes in all detectors, 60 demand scenarios were developed (5 weekdays times 12 months). However, some detectors were faulty and did not provide data in some locations or days. Also, due to the elimination of data with incidents or adverse weather, fewer days were analyzed. To account for the missing data, volumes from the next or previous months were used.

In some locations, on-ramp and off-ramp volumes were not available for the entire year. Flow balancing was attempted to calculate on-ramp or off-ramp volumes where data were not available, but in some cases, the results showed big discrepancies, so the attempt was aborted. The missing volumes were eventually obtained from the original VISSIM model that was given by KDOT. The locations that data were assumed were at Southwest Blvd off-ramp, 18th St on and off ramps, and Lamar off-ramp.

5. CALIBRATION AND SIMULATION

In this chapter the process that was conducted to create the VISSIM file and run the simulations is discussed. Modeling the facility is discussed first. Next, the selection of the vehicle characteristics (acceleration, deceleration, etc.) is presented. After that, the calibration of the VISSIM model is discussed. Then, running the simulation model is explained. Lastly, the coding of the ramp metering algorithms in VISSIM is provided.

5.1. Modeling the Facility

The geometric modeling of the facility in VISSIM consists of creating links and connectors. Links are the main components of the roadway as they represent the path the vehicles take to travel through the roadway. An initial file with the geometry of the facility was provided by KDOT. All links in the VISSIM model are modeled to have 12ft lane width and the number of lanes that correspond to the field. VISSIM provides a background map for the study location. The background map was used to draw the links of I-35 southbound from Cambridge Circle to 75th St, including on-ramps and off-ramps. Some on-ramps had traffic signals on the arterial streets. These traffic signals were removed and the links from the arterials were joined together and the joined links were extrapolated beyond the on-ramp storage for 2000ft. This was done because the signal timing and the volume distribution on the arterial is unavailable. The connectors are used to connect links with each other if there is a reduction or addition in the number of lanes, on-ramps or off-ramps, or if a change in link behavior in a particular location is needed. Screenshots of the I-35 corridor (from Google.Maps.com) and the VISSIM model are shown in Figure 8 and Figure 9, respectively.



Figure 8 Satellite screenshot of I-35 corridor (maps.google.com)

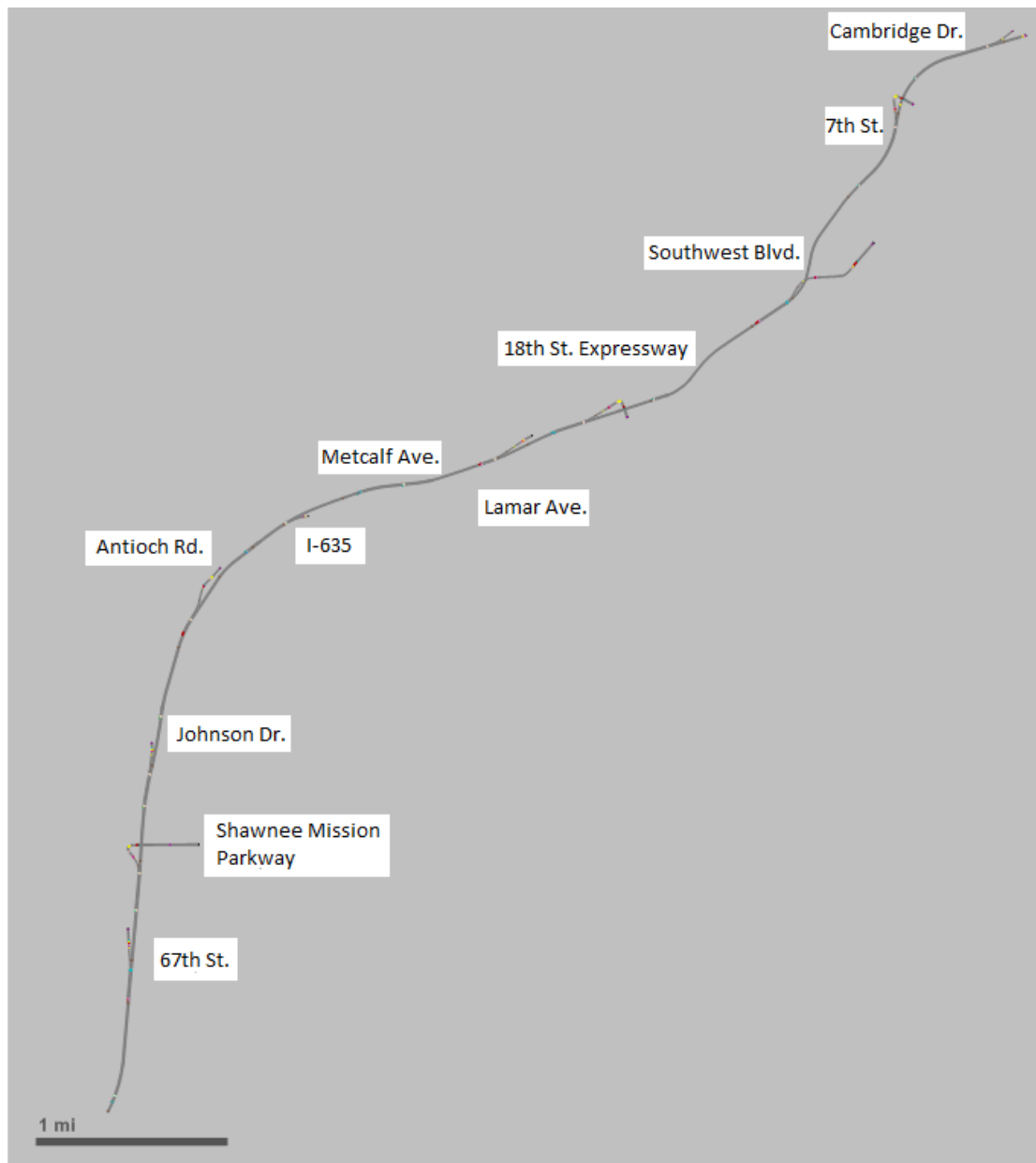


Figure 9 VISSIM model of I-35 corridor

5.2. Vehicle Properties

The VISSIM model provided by KDOT was used to obtain the percentage of passenger cars, Heavy Goods Vehicle (HGV), and Light-Heavy Goods Vehicle (L-HGV). As for the desired speed, the mainline detectors speed readings obtained from KC Scout portal were reviewed and a mean speed of 64 mi/hr was identified. It was assumed that the desired speed distribution follows an S-shaped curve. Figure 10 shows the desired speed distribution created for the VISSIM model.

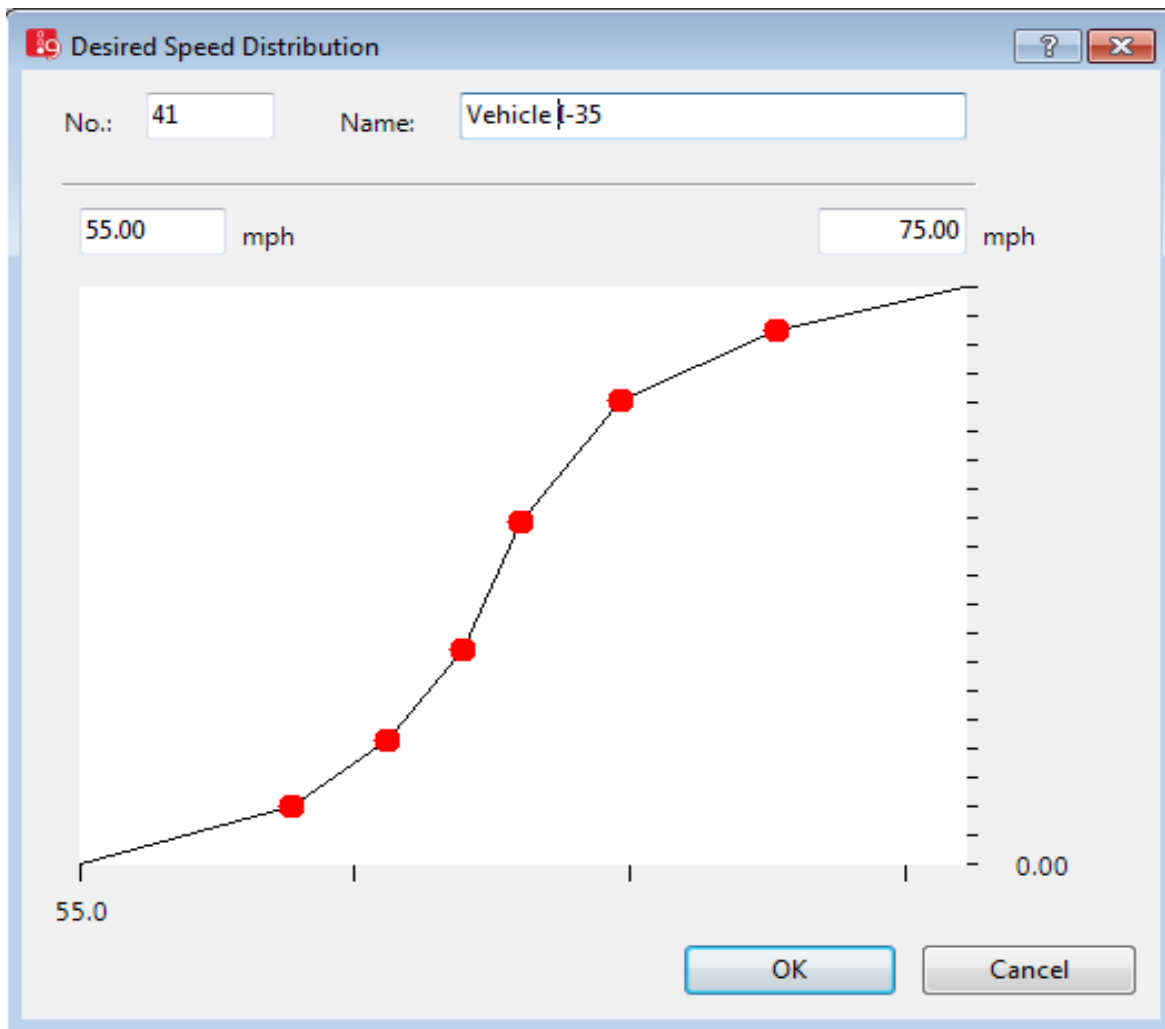


Figure 10 Desired Speed Distribution

For the maximum and desired acceleration and deceleration rates, the default distributions provided by VISSIM were used, as the exact values were not available.

5.3. Calibration

The goal of calibrating the VISSIM model was to create a model in which real life traffic can be replicated accurately. Once the calibration was completed, the model was used to test the effects of ramp metering on traffic operations. To calibrate the simulation model, the average speed throughout the analysis period was the selected performance measure. The average speed from the simulation had to match the average speed based on real life data within 10%. The VISSIM model

was calibrated by changing the driver behavior parameters in the links and connectors. A calibration day without traffic incidents or adverse weather conditions in all its detectors was chosen. The calibration day is April 22nd 2016.

The calibration was conducted by changing the values of multiple parameters throughout the software. To compare the simulation model to the calibration day, detectors were established in the VISSIM model in the same locations they exist in real life. Through the simulation animation unrealistic driver behaviors (e.g., freeway vehicles stopping in the middle lane unable to exit to the off-ramp, etc.) were identified, and were resolved with the help of the calibration. A trial and error process was undertaken, where the simulation model would run once each time a calibration parameter would change, and then the speeds were evaluated until a similar speed profile was achieved. Once the speeds between simulation and field data were in close match, the simulation model was run multiple times with different seed numbers and the average speed profile was again compared to that of the calibration day. To accomplish the calibration, the vehicle routes decision and connectors' lane change distance, and driving behavior parameters were used.

5.3.1. Connectors Lane Change Distance and Vehicle Routes Decision

When a vehicle route segment starts, the destination is programmed for each vehicle. Thus, vehicle routes need a sufficient distance before exiting the freeway to allow vehicles enough distance to move to the shoulder lane. Vehicles need at least 2,100ft before an off-ramp to maneuver successfully without blocking the mainline (Leyn and Vortisch 2015). To optimally model vehicle route decisions on the I-35 corridor based on KC Scout Portal volume data, the start point should be immediately after an on-ramp link ends (on the mainline link) and end before the next on-ramp. This method did not yield a realistic driver behavior in some cases due to small distances between an on-ramp and the next off-ramp. The small distance causes some vehicles

traveling on the left-most lane not to maneuver successfully to the shoulder lane thus blocking the mainline. To resolve this issue, vehicles entering from some on-ramps were assumed to not exit on the next off-ramp or the next two off-ramps.

Connectors' lane change distance was adapted from Leyn and Vortisch (2015) and simulation trials. According to Leyn and Vortisch (2015) the lane distance for an off-ramp diverge should be at least 700m (2,100ft). The suggested value was used in the initial runs, and it was adjusted for the next test runs so that the vehicle maneuverability looked realistic and the average speeds from the simulation matched to the calibration day speeds.

5.3.2. Driving Behavior

Driving behavior assigns the aggression, and awareness of vehicles in a link. According to Leyn and Vortisch (2015), driver behavior parameters differ by segment type (basic freeway, diverge, merge, and weave). This thesis used the same values in Leyn and Vortisch (2015) to calibrate the VISSIM model in the initial run. After that, the car following, and lane change parameters were adjusted to produce similar speeds to the calibration day.

Table 6 Calibrated Wiedemann 99 model parameters

Parameter	Default	Freeway	Diverge	Merge	Weave
CC0	4.92	5.60	7.50	7.80	9.00
CC1	0.90	1.05	1.15	1.30	1.45
CC2	13.12	13.10	12.00	12.00	12.60
CC3	-8.00	-8.00	-8.00	-8.00	-8.00
CC4	-0.35	-0.30	-0.35	-0.35	-0.35
CC5	0.35	0.30	0.35	0.35	0.35
CC6	11.44	11.44	11.44	11.44	11.44
CC7	0.82	0.82	0.82	0.82	0.82
CC8	11.48	10.48	10.48	12.48	12.48
CC9	4.92	4.92	4.92	4.92	4.92

Table 7 Calibrated lane change parameters

	Default	Freeway	Diverge	Merge	Weave
General Behavior	Free Lane selection	Free Lane selection	Free Lane selection	Free Lane selection	Free Lane selection
Max. Deceleration (ft./s ²)					
Own	-13.12	-13.12	-13.12	-13.12	-13.12
Trailing Vehicle	-9.84	-9.84	-9.84	-9.84	-12.00
-1 ft./s ² per distance (ft.)					
Own	100	300	200	300	400
Trailing Vehicle	100	200	200	200	400
Accepted deceleration					
Own	-3.28	-3.00	-3.00	-4.50	-4.50
Trailing Vehicle	-3.28	-2.25	-3.00	-3.00	-4.50
Waiting before diffusion (s)	60	60	60	60	60
Min. headway (front/rear)	1.64	1.64	1.64	1.64	1.64
Safety distance reduction factor	0.60	0.60	0.7	0.45	0.4
Max. deceleration for cooperative braking (ft./s ²)	-9.84	-9.84	-16.00	-18.00	-18.00
Advanced Merging	✓	✓	✓	✓	✓
Cooperative lane change	✓	✓	✓	✓	✓
Max. speed difference	1.84	16.00	16.00	16.00	16.00
Max. collision time	10.00	10.00	10.00	10.00	10.00

5.3.3. Calibration Results

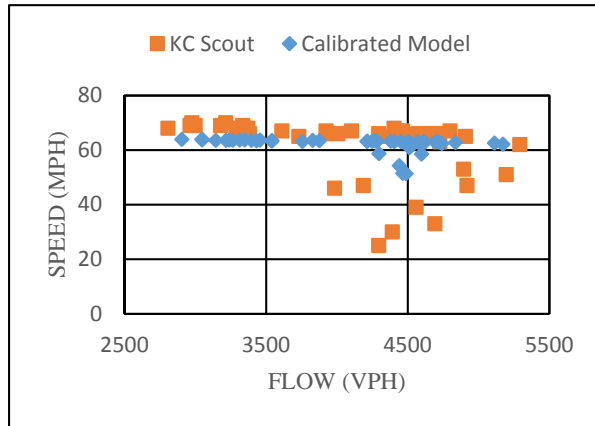
For the calibration process, ten random seed numbers that produced speed heat maps (inspected visually) close to the actual speed heat map were selected, after initially testing 150 random seeds. A speed heat map by location is shown for both field-measured and simulated data in Figure 11 and Figure 12, respectively. According to Figure 11, in real life there should be congestion at the Antioch St. upstream and Antioch St. downstream merge detectors, however, from calibration it can be seen that there is no congestion there. In addition, congestion events at Johnson Dr. and Shawnee Mission Parkway occur later than in real life. Unfortunately, it was not possible to match these speeds during calibration.

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
Time of Day	Cambridge Dr.	7th St.	SW Blvd. Far US	SW Blvd. US	SW Blvd. DS Merge	18th St.	Lamar Ave. Far US	Lamar Ave. US	Metcalf Ave. US	I-635 DS	Before Antioch St.	Antioch St. US	Antioch St. DS Merge	Johnson Dr. US	SMP US	67th St. US	67th St. DS Merge
15:00:00	61	62	65	60	64	64	64	55	62	57	65	65	64	62	60	60	58
15:05:00	63	64	66	60	62	63	63	55	63	58	62	64	62	62	59	58	60
15:10:00	61	63	68	60	63	64	62	53	63	58	63	64	61	59	58	53	59
15:15:00	60	62	65	53	62	63	62	53	61	57	62	63	61	60	51	44	49
15:20:00	59	62	65	53	60	63	60	51	62	59	62	64	62	61	47	36	46
15:25:00	62	63	66	58	51	62	63	54	63	58	63	64	61	60	45	31	42
15:30:00	61	63	67	61	62	63	53	48	63	58	63	65	63	60	44	30	43
15:35:00	60	63	66	60	60	63	50	41	61	59	64	66	64	61	44	30	39
15:40:00	62	62	66	60	61	60	60	52	61	55	61	62	61	60	42	26	38
15:45:00	61	62	67	60	62	64	62	54	63	59	63	65	61	40	34	31	37
15:50:00	61	62	67	61	62	56	60	52	62	56	64	64	63	37	29	22	26
15:55:00	62	63	67	62	64	64	62	54	64	60	64	66	55	23	25	30	42
16:00:00	61	63	65	61	63	65	64	55	64	60	65	64	31	23	28	35	43
16:05:00	59	62	66	60	62	65	63	53	62	60	65	64	51	30	27	26	35
16:10:00	60	62	62	58	56	45	57	49	61	60	62	65	61	26	26	29	35
16:15:00	58	61	51	32	46	38	53	45	60	58	65	63	35	24	24	25	38
16:20:00	57	60	30	22	37	53	57	46	59	56	61	53	24	22	22	24	35
16:25:00	61	45	33	34	45	44	55	49	58	52	52	25	29	26	25	26	35
16:30:00	62	63	47	19	37	56	39	31	58	54	55	35	24	19	21	25	32
16:35:00	61	63	39	22	43	45	36	30	56	56	46	26	21	22	21	22	34
16:40:00	61	63	53	27	38	36	40	39	58	47	42	25	25	23	23	25	37
16:45:00	59	60	46	18	36	45	54	44	58	46	40	30	26	20	22	24	36
16:50:00	62	61	25	19	44	60	55	41	59	52	52	30	21	20	23	27	34
16:55:00	63	64	47	24	47	58	58	50	59	57	51	23	22	19	21	24	36
17:00:00	61	62	66	56	48	59	60	52	57	47	42	28	26	21	25	30	39
17:05:00	61	62	65	58	62	63	61	53	62	58	55	26	23	28	29	31	34
17:10:00	62	62	65	60	62	62	59	52	61	58	62	52	39	26	24	23	33
17:15:00	61	63	65	59	62	62	62	54	61	60	62	56	33	21	23	26	39
17:20:00	59	61	63	59	60	63	61	53	61	57	61	61	27	26	25	26	37
17:25:00	62	64	67	61	63	63	61	52	59	56	59	61	35	27	25	26	31
17:30:00	63	64	67	62	64	65	64	55	63	60	63	65	40	27	24	22	31
17:35:00	61	63	66	61	64	65	65	57	64	61	65	65	57	23	25	28	36
17:40:00	64	64	70	63	66	65	63	55	62	59	62	66	63	41	32	24	31
17:45:00	62	65	68	62	65	66	64	56	63	60	60	65	64	61	43	27	33
17:50:00	62	65	69	63	64	66	63	55	63	60	64	65	63	62	44	28	34
17:55:00	62	63	68	61	64	65	64	56	64	62	64	66	64	63	60	57	55
18:00:00	63	64	69	63	66	67	66	56	64	60	65	66	64	61	60	61	62
18:05:00	63	65	69	62	65	65	66	56	65	62	66	68	66	64	62	62	62
18:10:00	63	65	69	63	66	66	64	55	64	61	65	66	65	63	62	62	64
18:15:00	62	65	68	63	65	66	64	56	64	61	65	67	65	63	62	61	63
18:20:00	62	65	69	62	65	65	66	56	64	59	65	66	63	62	61	62	63
18:25:00	62	66	70	63	67	66	66	57	65	62	65	67	64	62	60	61	62
18:30:00	62	64	68	63	65	66	65	56	65	61	66	66	65	64	62	62	63

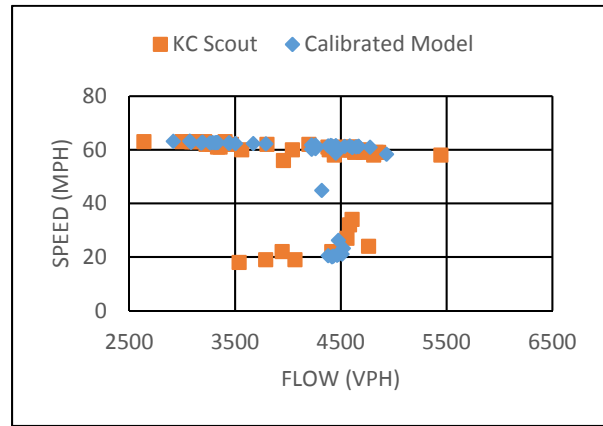
Figure 11 KC Scout speed profile

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
Time of Day	Cambridge Dr.	7th St.	SW Blvd. Far US	SW Blvd. US	SW Blvd. DS Merge	18th St.	Lamar Ave. Far US	Lamar Ave. US	Metcalf Ave. US	I-635 DS	Before Antioch St.	Antioch St. US	Antioch St. DS Merge	Johnson Dr. US	SMP US	67th St. US	67th St. DS Merge
15:00:00	62	63	63	62	60	62	62	60	63	61	61	62	62	62	63	62	55
15:05:00	62	63	64	62	58	62	62	60	63	61	61	62	61	62	62	62	54
15:10:00	61	63	63	62	56	62	61	59	63	61	61	62	61	62	63	62	47
15:15:00	61	63	63	61	52	61	60	50	62	60	60	62	60	62	62	55	44
15:20:00	61	63	63	61	53	62	52	37	62	61	60	62	61	62	62	41	38
15:25:00	61	63	63	61	52	62	52	37	62	61	60	62	60	62	62	30	38
15:30:00	62	63	63	62	56	62	46	32	62	61	61	62	60	62	61	30	36
15:35:00	61	63	63	62	56	62	39	26	62	61	61	62	60	62	62	26	35
15:40:00	61	63	63	61	55	61	38	30	62	60	60	62	60	62	48	23	36
15:45:00	61	63	63	61	53	62	45	31	62	61	61	62	60	62	35	22	36
15:50:00	61	63	63	59	49	61	54	42	62	61	61	62	59	61	32	22	36
15:55:00	61	63	63	60	54	60	56	53	62	61	61	62	60	55	21	22	36
16:00:00	61	63	63	60	54	59	57	47	62	61	60	62	60	53	20	23	36
16:05:00	61	63	63	62	54	58	54	44	62	61	61	62	59	50	20	22	36
16:10:00	60	62	62	58	51	53	57	49	62	61	60	62	59	49	21	22	35
16:15:00	61	62	63	26	46	34	57	52	62	60	60	62	59	44	20	22	35
16:20:00	60	62	62	21	47	28	57	56	62	61	60	62	59	32	20	22	36
16:25:00	61	62	61	20	45	26	57	53	62	61	60	62	58	30	20	22	35
16:30:00	61	63	59	20	46	26	56	49	62	61	60	62	60	24	20	22	36
16:35:00	61	63	51	21	47	26	55	47	62	61	61	62	59	23	20	22	35
16:40:00	61	63	54	21	46	26	54	49	63	61	61	62	60	23	20	22	35
16:45:00	61	63	52	21	46	28	56	51	62	61	61	62	59	23	20	22	35
16:50:00	61	63	59	21	45	29	55	51	62	61	60	62	60	23	20	22	35
16:55:00	62	63	63	23	47	31	57	50	62	62	61	62	59	24	19	22	35
17:00:00	61	63	63	45	50	33	58	47	62	61	61	63	60	25	20	22	35
17:05:00	61	63	63	60	54	47	56	39	62	61	61	62	60	24	20	22	35
17:10:00	61	63	63	61	54	55	56	44	62	61	61	62	58	23	20	22	36
17:15:00	61	62	63	61	55	60	59	48	63	61	60	62	52	20	20	22	34
17:20:00	60	62	63	61	52	61	59	53	62	61	61	62	53	22	20	22	34
17:25:00	61	63	63	61	54	62	58	56	62	61	61	62	54	20	19	22	35
17:30:00	62	63	64	62	60	62	60	57	63	61	61	62	54	20	20	22	36
17:35:00	62	63	64	63	60	63	62	61	63	61	61	62	56	24	21	22	35
17:40:00	62	63	63	62	60	62	62	61	63	61	62	63	61	40	20	22	36
17:45:00	62	63	64	63	60	62	62	61	63	62	61	63	61	50	20	22	35
17:50:00	62	64	64	63	62	63	62	61	63	62	62	63	62	61	28	22	36
17:55:00	62	63	64	63	60	63	62	61	63	62	62	63	62	63	51	25	36
18:00:00	62	63	64	63	60	63	62	61	63	62	62	63	62	63	63	51	46
18:05:00	62	63	64	63	60	63	62	61	63	62	62	63	62	63	63	63	56
18:10:00	62	63	64	63	60	63	62	60	63	62	62	63	62	63	63	63	58
18:15:00	62	63	64	63	61	63	62	61	63	62	62	63	62	63	63	63	57
18:20:00	62	63	64	63	60	63	62	61	63	62	62	63	62	62	63	62	56
18:25:00	62	63	64	63	61	63	63	61	63	62	62	63	62	63	63	62	56
18:30:00	62	63	64	63	62	63	62	61	63	61	62	63	62	63	63	63	56

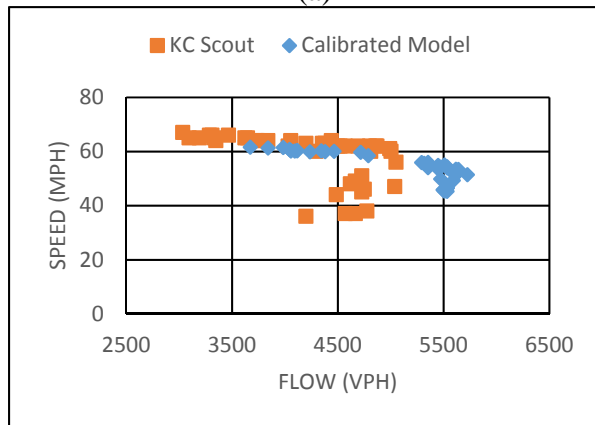
In addition, speed flow diagrams for the mainline detectors at the congested locations along the facility were considered for the calibration. Figure 14 (a-h) presents the speed flow diagrams using both KC Scout data and the calibrated data.



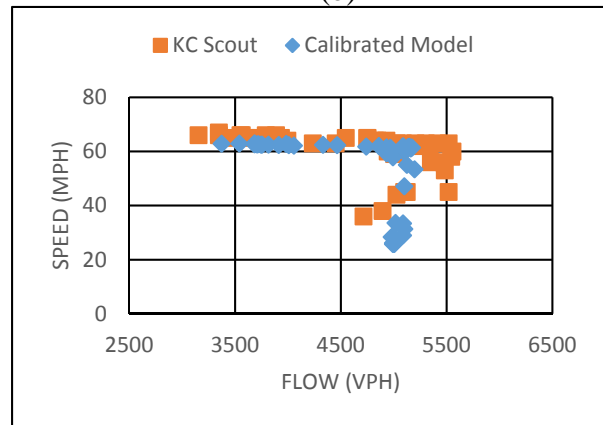
(a)



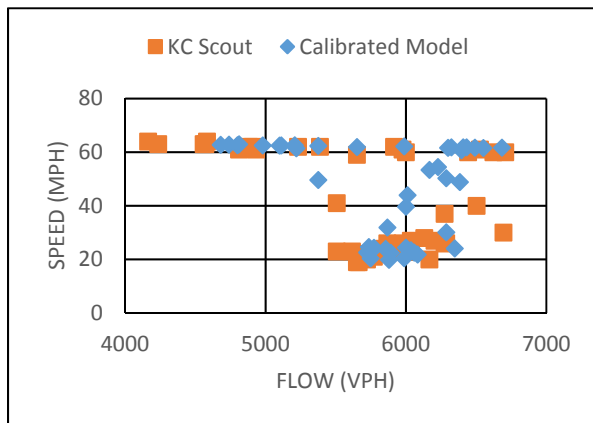
(b)



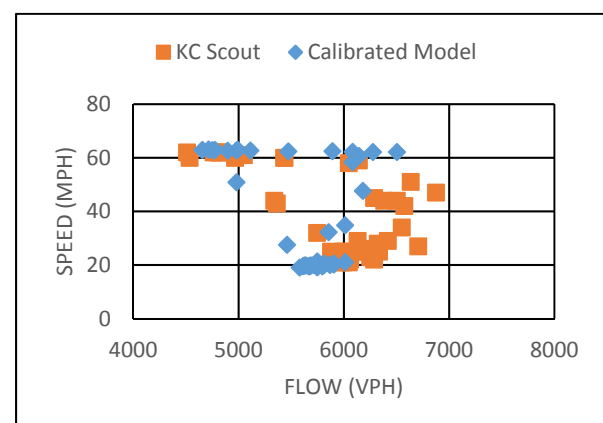
(c)



(d)



(e)



(f)

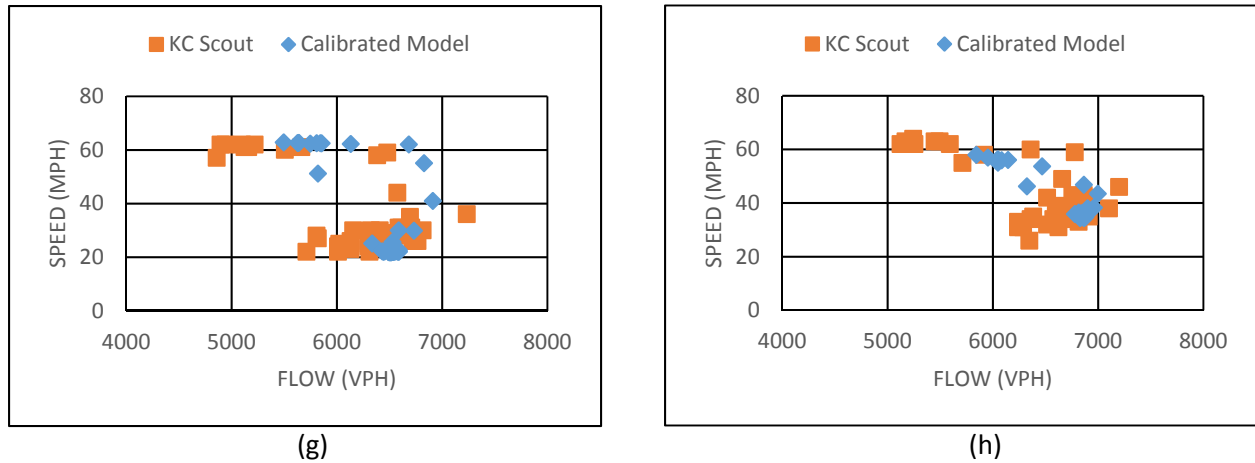


Figure 14 Speed flow diagram at (a) 7th St. merge, (b) SW Blvd. upstream, (c) SW Blvd. merge, (d) 18th St. upstream, (e) Johnson Dr. upstream, (f) Shawnee Mission Parkway (SMP) upstream, (g) 67th St. upstream, and (h) 67th St. merge detectors

From this figure it is shown that in the majority of the cases, the speed-flow curves are in good agreement.

5.4. Simulation of Control Scenarios

This section describes the steps taken to test No Control, ALINEA, and HERO control scenarios through simulation. 60 demand scenarios were tested for each of the three control scenarios. Each demand scenario was run using four random seeds, different from the other demand scenarios but the same for each control scenario. The No Control scenario was modeled with no ramp meters on the on-ramps. For ALINEA and HERO, the ramp meters were located on 7th St., SW Blvd, 18th St. and 67th St. on-ramps. The maximum number of queued vehicles at the metered sites is shown in Table 8. This queue capacity was estimated based on the geometry of the on-ramp, and was used for operating the metering strategies.

Table 8. Queue Capacity at Metering Locations

On-Ramp	7 th St.	SW Blvd.	18 th St.	67 th St.
Queue Capacity	24 Vehicles	70 Vehicles	28 Vehicles	14 Vehicles
Max Queue Length	300ft	950ft	375ft	350ft

5.4.1. ALINEA Algorithm

ALINEA is an occupancy-based local ramp metering algorithm. For ALINEA to be able to function properly the downstream mainline detector that measures occupancy should be placed where congestion starts. In real life, congestion at merge segments occurs downstream of the ramp gore area by a few hundred meters which is usually after the acceleration lane drop (Papageorgiou et al., 2007). However, in microsimulation, congestion at merge segments occurs between the ramp gore area and the acceleration lane drop. A simulation run was reviewed and the location of congestion was identified to occur before the acceleration lane drop by 300ft.

ALINEA uses the downstream detector critical occupancy to maximize the mainline throughput. Critical occupancy is the occupancy corresponding to the maximum vehicle throughput. The detailed description of this algorithm is presented in Chapter 2, but the equation used to model it into VISSIM is shown below:

$$r(k) = r(k - 1) + K_R[\hat{o} - o_{out}(k)] \quad \text{Equation 34}$$

Where $r(k)$ is the current ramp metering rate, $r(k - 1)$ previous iteration ramp metering rate, K_R is a regulator parameter (smoothing factor), \hat{o} is the desired downstream occupancy and o_{out} is the measured occupancy. K_R was considered to be 70 based on the literature (Papageorgiou et al., 1991).

Ten demand scenarios consisting of two demand scenarios for each weekday would run to determine the critical occupancy at these locations. The occupancy rate was collected at 30-second intervals. Figure 13 (a-d) presents downstream flow-occupancy graphs at selected locations. Figure 13 (a-d) was used to identify the critical occupancy at the locations downstream of the merge. These values are shown in Table 9 below.

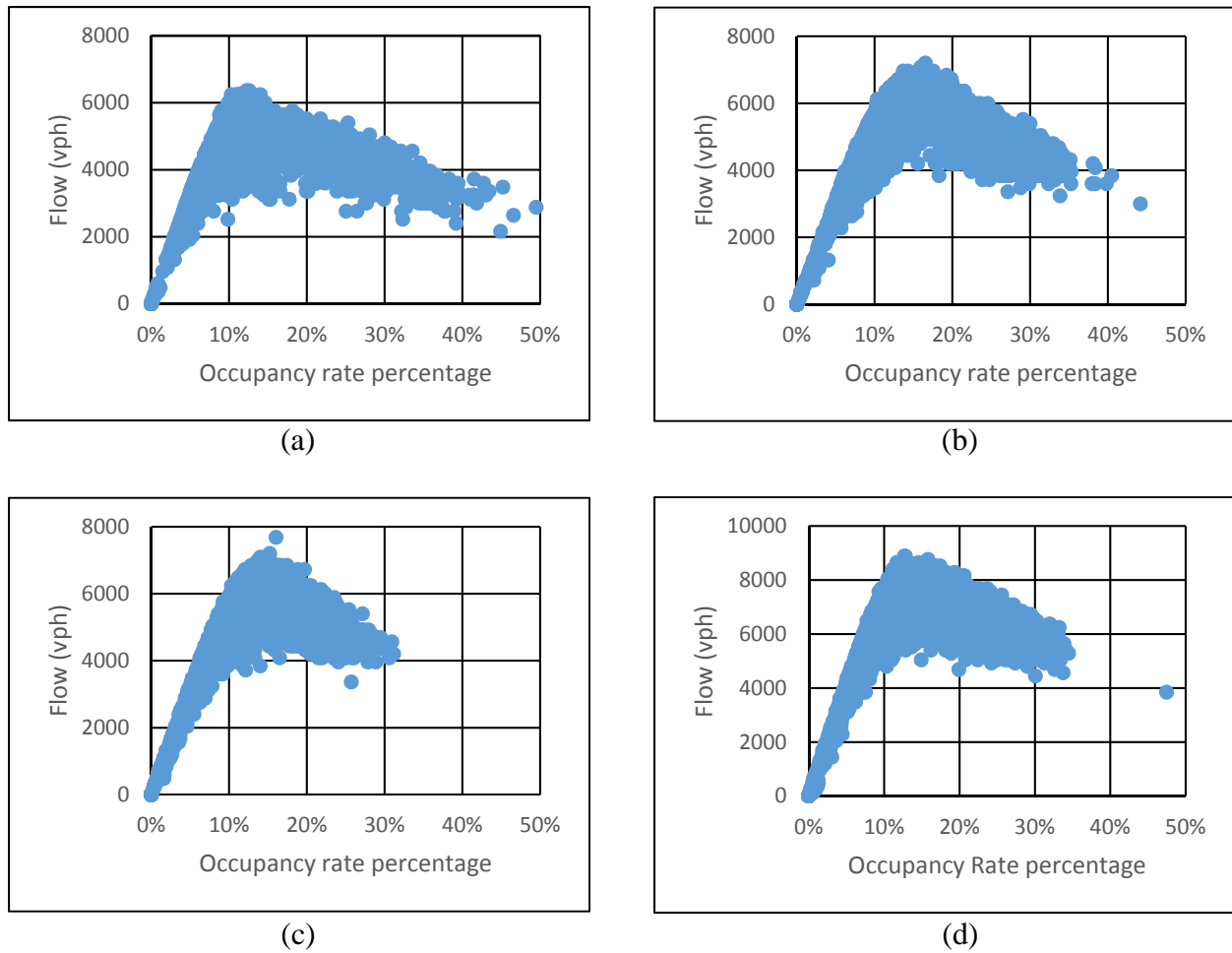


Figure 13 Flow occupancy rate diagram at (a) 7th St. merge, (b) SW Blvd. downstream, (c) 18th St. downstream, and (d) 67th St. downstream before the end of acceleration lane

Table 9. Critical occupancy downstream of the metered ramps

On-Ramp	7 th St.	SW Blvd.	18 th St.	67 th St.
Critical Occ. rate	13%	15%	15%	14%

An occupancy rate slightly less than the critical occupancy was used as the desired occupancy, \hat{o} , in ALINEA, to ensure that the traffic stays in free flowing conditions. An occupancy rate of 13% was used for all ramp meters. Also, the time step to calculate the cycle length was set to be 20 seconds at the 7th St, 18th St, and 67th St ramp meters and 30 seconds at the SW Blvd ramp

meter. The time step was calculated as the travel time a vehicle needs to reach from the ramp meter to the downstream detector. Ramp metering was activated at 3:00pm and if the occupancy rate is above 11% for two consecutive time steps, and terminate at 6:00pm. The ramp metering flow for all meters was set to range between 720 vphpl to 200 vphpl, which is equivalent to a ramp metering cycle length of 5 seconds to 9 seconds per vehicle, respectively. For queue flush, a ramp cycle length of 5 seconds was implemented when the queue length reaches 75% of the queue capacity on an on-ramp. The queue detectors are placed in the same location they are installed in the field.

In VISSIM, if a vehicle passes on two adjacent queue detectors at the same time, it is counted as two vehicles. To eliminate erroneous vehicle readings, the queue count was reset to zero if the ramp storage was empty and set to maximum queue length if the ramp storage was above queue capacity. The VisVAP flow chart for the ALINEA algorithm was originally derived from a dissertation (Armstrong 2011) and it was adjusted to match the needs of this thesis. The code for the ALINEA algorithm is presented in the Appendix A.

5.4.2. HERO Algorithm

As mentioned in the literature (Chapter 2) HERO is a system-wide algorithm that uses the ALINEA algorithm in addition to an on-ramp master slave configuration. HERO was implemented on 18th St and SW Blvd with 18th St acting as the master ramp, and SW Blvd and 7th St with SW Blvd acting as the master ramp. The master on-ramp starts using the slave queue storage when its queue reaches 50%, and the storage size that is used in the slave on-ramp is 50%. For example, if the queue on 18th St on-ramp reaches 16 vehicles, the slave on-ramp will start creating a queue of 35 vehicles if it was less than 35. The queue activation percentage (50%) on the master ramp and the queue creation on the slave ramp are user-defined. Any percentage an operator wants could be implemented. The logic behind using 50% activation and queue creation is that the ramps are

located relatively far away from each other. Therefore, for the slave on-ramp queue creation to take effect, the master on-ramp queue still has time before it reaches capacity. In addition, in real life there are traffic signals on the arterials connecting to the on-ramp, so if a signal turns green it will supply the on-ramp with a large number of vehicles in a small amount of time. This demand must have enough storage on the on-ramp to avoid overflowing into the arterial. The code for the HERO algorithm is presented in the Appendix B.

6. SIMULATION RESULTS

This section discusses the simulation results of the No Control, ALINEA, and HERO control scenarios. Travel time, speed, throughput, queue length, and travel time reliability were the selected performance measures to evaluate the three control scenarios. Data of the performance measures were collected in 5-minute intervals between 3:00pm to 6:30pm. The average of every 5-minute interval in the 60 demand scenarios was used for the analysis and comparison.

6.1. Travel Time

Four different travel time-related performance measures were evaluated in this thesis. The first one is the facility travel time, which is considered as the travel time of vehicles traveling from Cambridge Circle junction to 75th St. junction. The entire facility was split into two parts: the first (northern) part, and the second (southern) part. Travel time for all vehicles traveling along the first part (om Cambridge Circle junction to Metcalf Ave off-ramp) and the second part (Metcalf Ave. to 75th St. junction) was also evaluated. Lastly, the queue travel time on the metered on-ramps was also estimated. The exact location of the facility and all junctions is shown in Figure 9.

6.1.1. Facility Travel Time and Travel Time Reliability

Facility travel time measures the travel time (TT) of vehicles that traveled the entire facility from Cambridge Circle junction to 75th St junction. The free flow TT of the entire facility was measured to be 465 seconds. Figure 14 presents the Travel Time Index TTI with respect to time of day from 3:00pm (0 sec) to 6:30pm (12,600 sec) for the three control scenarios.

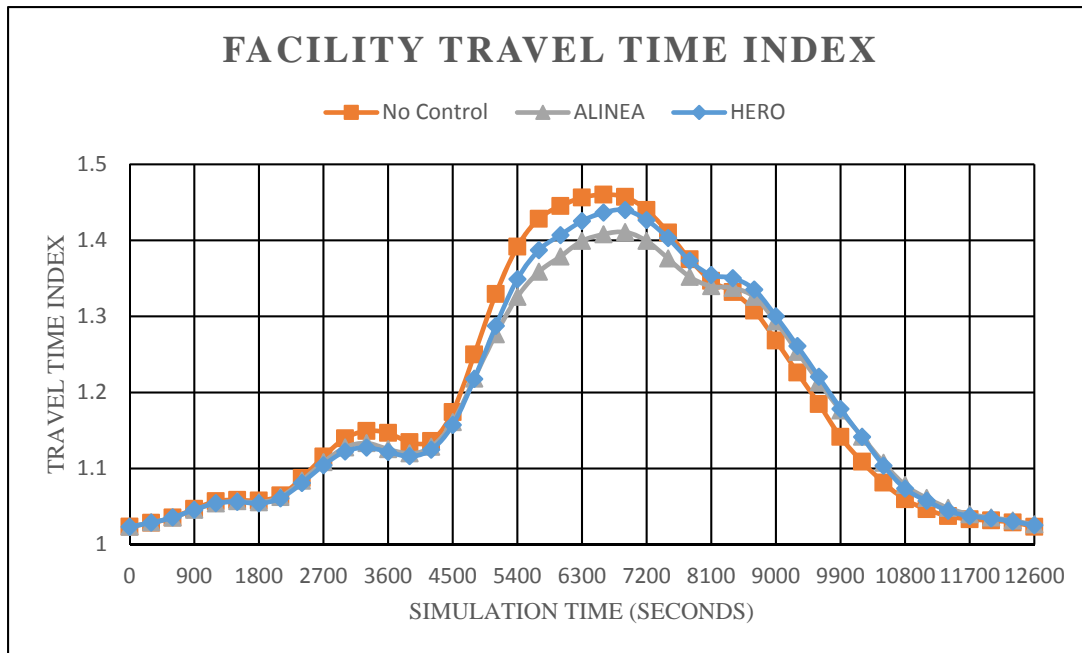


Figure 14 Facility Travel Time Index for the three control scenarios

According to Figure 14, HERO has lower TTI than the No control scenario from 2700 second (3:45pm) to 7800 second (5:10pm). ALINEA has lower TTI than the No Control scenario from 2700 second (3:45pm) to 8400 second (5:20pm). However, towards the end of the simulation the No Control scenario TTI becomes lower than HERO and ALINEA. This can be attributed to the effect of ramp meters. The ramp meters initially hold vehicles on the on-ramps, which leads to lower TTI on the mainline. Later on, when the ramp queue storage reaches capacity and starts discharging vehicles at 720phpl (higher than the on-ramp demand), mainline traffic conditions deteriorate and the freeway TTI increases.

Travel time reliability analysis was conducted for all 240 demand scenarios. The cumulative distribution of the facility TT is shown in Figure 15. From Figure 15 it can be depicted that ALINEA and HERO travel times are slightly less than the No Control scenario. The mean, median, 85th percentile, and 95th percentile TT of all three control scenarios are presented in Table 10. The percent difference in travel time, buffer time, and buffer index is presented in Table 11.

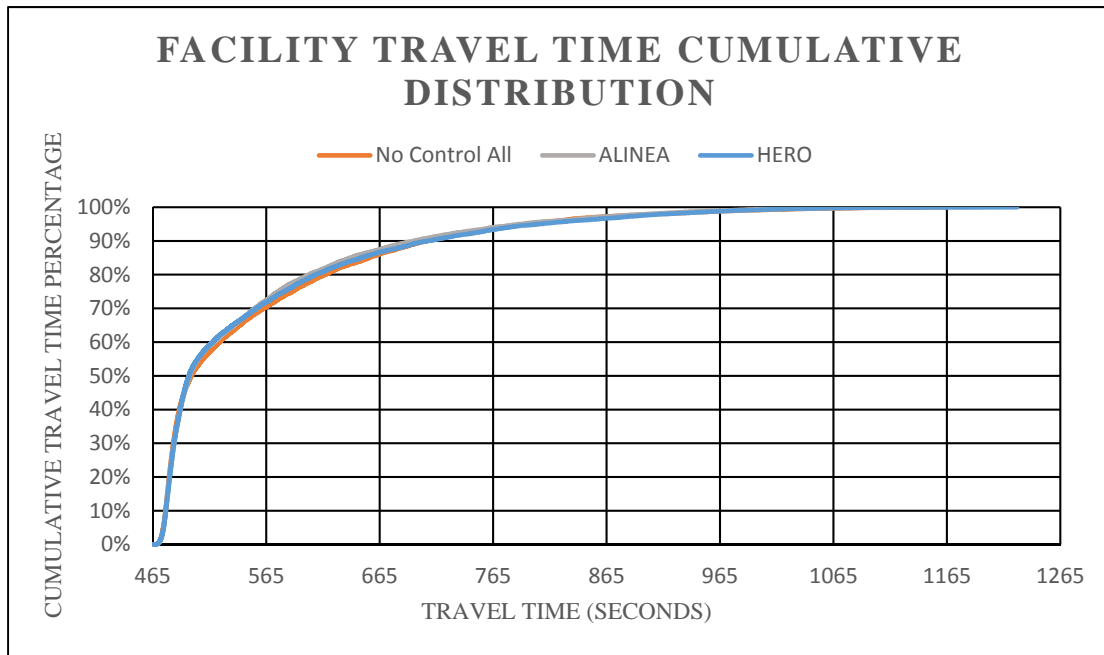


Figure 15 Facility Travel Time cumulative distribution

Table 10 Mean, median, 85th percentile, and 95th percentile for Facility Travel Time

	TT _{Mean}	TT _{Median}	TT ₈₅	TT ₉₅
No Control	553 sec	499 sec	655sec	795 sec
ALINEA	549 sec	497 sec	639 sec	789 sec
HERO	552 sec	497 sec	648 sec	805 sec

Table 11 Facility travel time, buffer time, and buffer index of the three control scenarios

	TT _{Mean} %Change	TT _{Median} %Change	TT ₈₅ %Change	TT ₉₅ %Change	Buffer Time BT	Buffer Index BI
No Control	-	-	-	-	242 sec	43.7%
ALINEA	-0.8%	-0.3%	-2.5%	-0.8%	240 sec	43.7%
HERO	-0.3%	-0.4%	-1.1%	1.3%	254 sec	46.0%

Tables 10 and 11 show the facility travel time results. It can be seen that ALINEA and HERO had lower TT_{Mean}, TT_{Median}, and TT₈₅ compared to the No Control scenario. However, HERO had a higher TT₉₅ and ALINEA had a lower TT₉₅ compared to No Control scenario. HERO had a higher buffer index while ALINEA had the same buffer index compared to No Control scenario.

6.1.2. First Half Travel Time and Travel Time Reliability

The first half travel time measures the travel time (TT) of vehicles that traveled from Cambridge Circle junction to Metcalf Ave. off-ramp (Figure 9). The free flow TT of the first half was measured to be 248 seconds. Figure 16 presents the Travel Time Index TTI with respect to time of day from 3:00pm (0) to 6:30pm (12,600) for the three control scenarios.

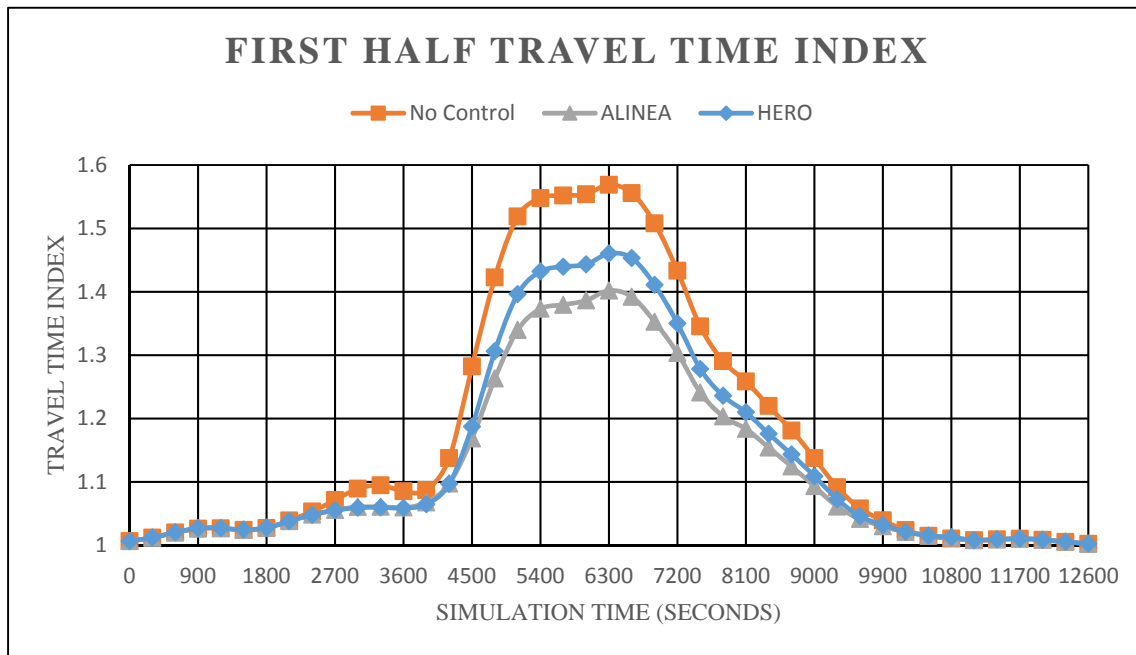


Figure 16 First Half Travel Time Index for the three control scenarios

According to Figure 16, HERO and ALINEA have lower TTIs than the No control scenario from 2100 second (3:35pm) to 9000 second (5:30pm). Also, ALINEA has lower TTIs than HERO. This is because ALINEA has higher queue length and total wait time on the on-ramps than HERO. The queue flushing system that ALINEA uses is less adaptable to large queues than HERO's queue control. More information about the queue length and waiting time is provided in section 6.1.4.

Travel time reliability analysis was conducted for all 240 demand scenarios. The First Half Travel Time cumulative distribution is shown in Figure 17. From Figure 17 it can be depicted that ALINEA and HERO Travel Times are less than the No Control scenario. The mean, median, 85th

percentile, and 95th percentile TT of all three control scenarios are presented in Table 12. The percentage difference in travel time, buffer time, and buffer index are presented in Table 13.

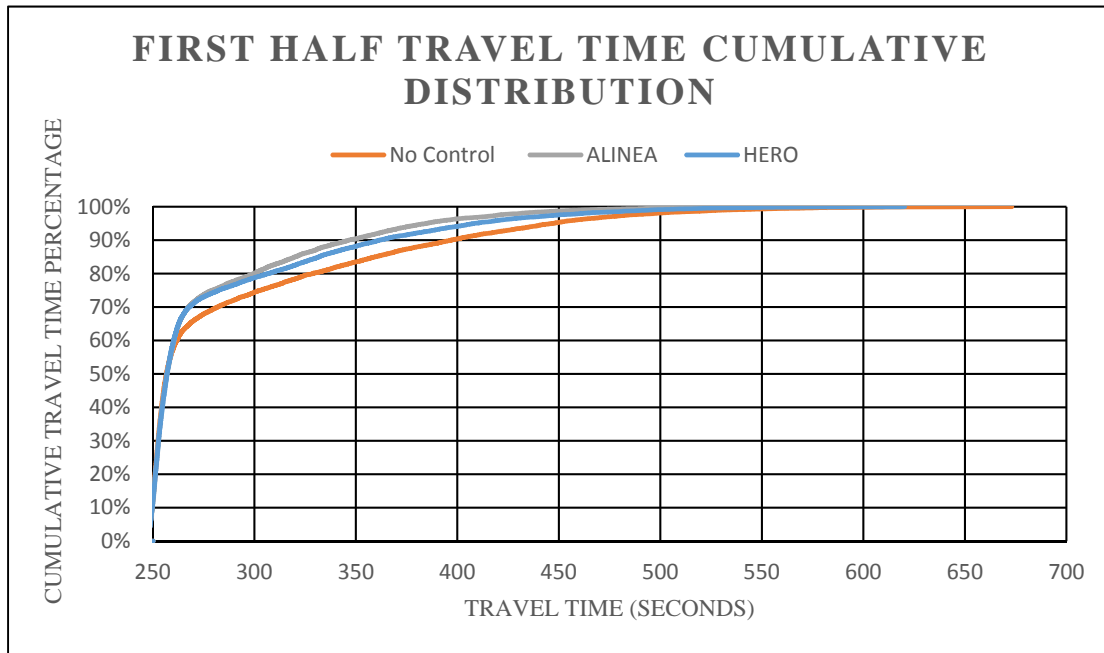


Figure 17 First Half Travel Time cumulative distribution

Table 12 Mean, median, 85th percentile, and 95th percentile for First Half Travel Time

	TT _{Mean}	TT _{Median}	TT ₈₅	TT ₉₅
No Control	291 sec	257 sec	359sec	447 sec
ALINEA	278 sec	257 sec	320 sec	384 sec
HERO	282 sec	257 sec	332 sec	408 sec

Table 13 First Half travel time, buffer time, and buffer index of the three control scenarios

	TT _{Mean} %Change	TT _{Median} %Change	TT ₈₅ %Change	TT ₉₅ %Change	Buffer Time BT	Buffer Index BI
No Control	-	-	-	-	156 sec	53.5%
ALINEA	-4.4%	0%	-11.0%	-13.9%	106 sec	38.2%
HERO	-3.1%	0%	-7.7%	-8.8%	126 sec	44.6%

Tables 12 and 13 show the first half travel times results. ALINEA and HERO had lower TT_{Mean}, TT₈₅, and TT₉₅ compared to the No Control scenario and there was no change in the TT_{Median} value. ALINEA and HERO have a lower buffer index compared to the No Control

scenario. In addition, ALINEA was superior compared to HERO in the first half in travel time and travel time reliability.

6.1.3. Second Half Travel Time and Travel Time Reliability

The second half travel time measures the travel time (TT) of vehicles that traveled from Metcalf Ave off-ramp to 75th St junction (Figure 9). The free flow TT of the second half was measured to be 214 seconds. Figure 18 presents the Travel Time Index TTI with respect to time of day from 3:00pm (0) to 6:30pm (12,600) for the three control scenarios.

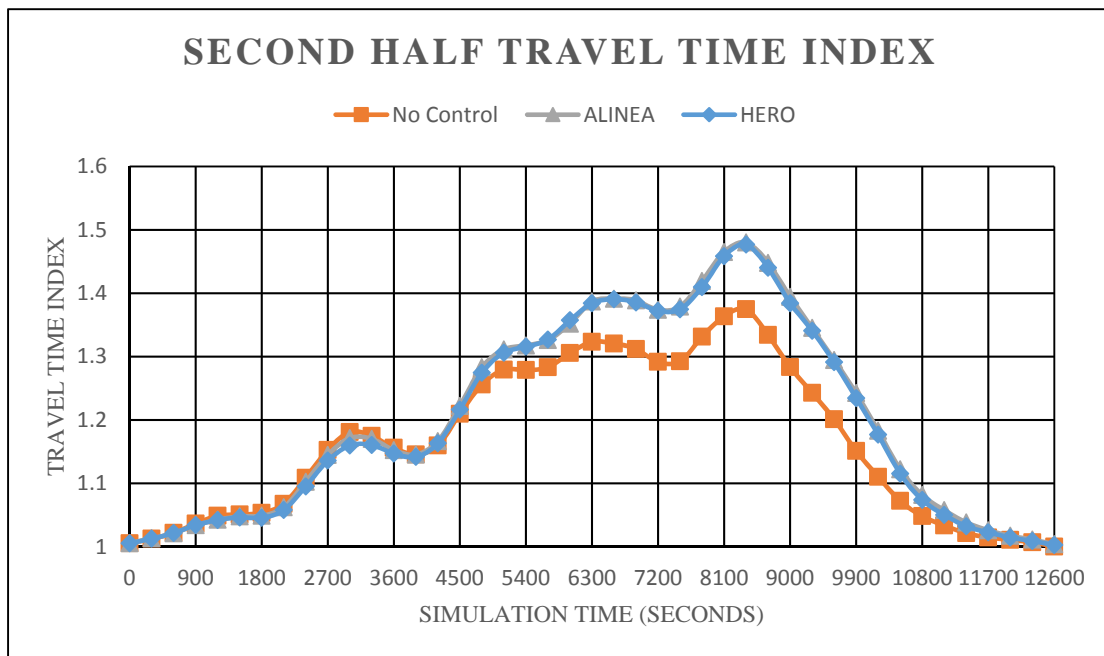


Figure 18 Second Half Travel Time Index for the three control scenarios

According to Figure 18 HERO and ALINEA have slightly lower TTI than the No control scenario from 900 second (3:15pm) to 3600 second (4:00pm). However, after 4500 second (4:15pm) the No Control scenario has lower TTI than ALINEA and HERO. This could be attributed to the lower travel times and higher speeds in the upstream segment (First Half). The ramp meters in the First Half lessen the effect of the bottlenecks of that segment, but this causes

vehicles to travel faster downstream and reach the bottleneck at 67th St. The ramp meter on 67th St. on-ramp improves the mainline TTI slightly from 3:15pm to 4:00pm as shown in Figure 18, but that effect diminishes afterwards.

Travel time reliability analysis was conducted for all 240 demand scenarios. The Second Half Travel Time cumulative distribution is shown in Figure 19. From Figure 19 it can be seen that the No Control scenario is better than ALINEA and HERO. The mean, median, 85th percentile, and 95th percentile TT of all three control scenarios are presented in Table 14. The percentage difference in travel time, buffer time, and buffer index are presented in Table 15.

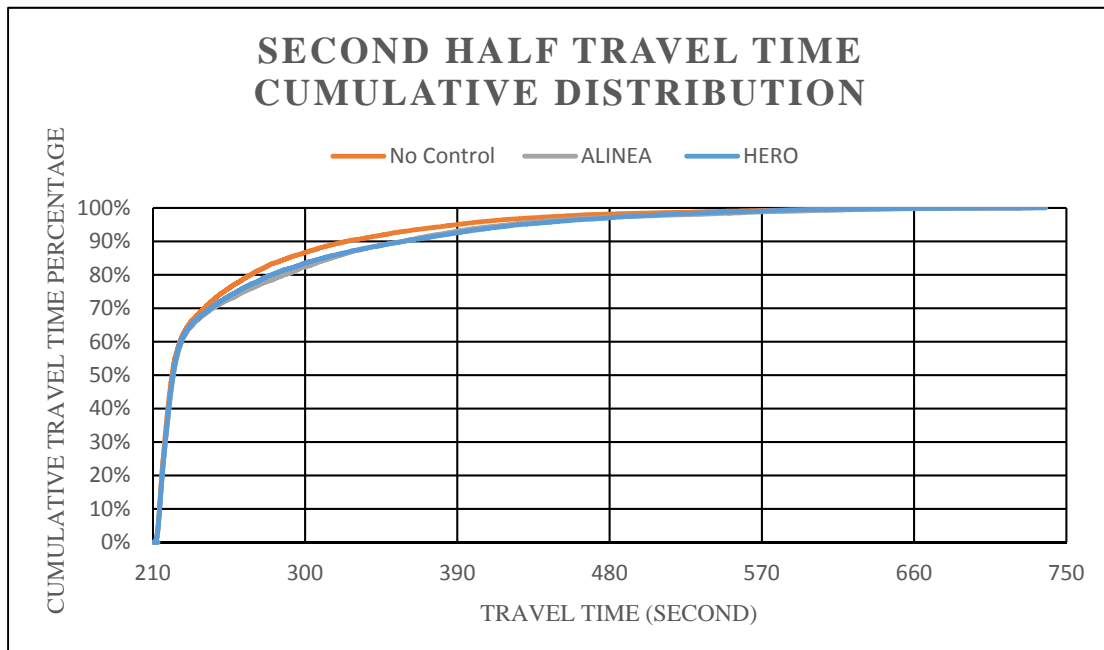


Figure 19 Second Half Travel Time cumulative distribution

Table 14 Mean, median, 85th percentile, and 95th percentile for second half travel time

	TT _{Mean}	TT _{Median}	TT ₈₅	TT ₉₅
No Control	249 sec	221 sec	290sec	390 sec
ALINEA	257 sec	222 sec	315 sec	420 sec
HERO	256 sec	222 sec	310 sec	426 sec

Table 15 Second half travel time, buffer time, and buffer index of the three control scenarios

	TT _{Mean} %Change	TT _{Median} %Change	TT ₈₅ %Change	TT ₉₅ %Change	Buffer Time BT	Buffer Index BI
No Control	-	-	-	-	140 sec	56.2%
ALINEA	3.1%	0.3%	8.8%	7.2%	163 sec	63.2%
HERO	2.7%	0.3%	7.2%	9.4%	170 sec	66.3%

Tables 14 and 15 show the second half travel times results. ALINEA and HERO had higher TT_{Mean}, TT_{Median}, TT₈₅, and TT₉₅ compared to No Control scenario. ALINEA and HERO have a higher buffer index compared to No Control scenario. In addition, ALINEA was superior compared to HERO in the second half in TT₉₅ and travel time reliability, however, HERO was better than ALINEA in TT_{Mean}, TT_{Median}, and TT₈₅.

6.1.4. Metered Ramps Queue Length and Total Waiting Time

When ramp meters are activated, they create queues on the on-ramp to reduce the demand entering the freeway. A ramp metering strategy may provide better mainline conditions but large queues that may interfere with the arterials may appear on the on-ramp. ALINEA uses the queue flushing strategy by setting the metering rate to maximum flow. HERO uses a queue control strategy that adapts for the current queue length present each time step, and sets a metering rate to optimize the ramp storage and prevent the queue from spilling back to the arterials. The queue length and waiting times from the simulation are shown in Table 16. Spillback % is the percentage of 5-minute intervals that experience queue length longer than ramp queue capacity.

Table 16 Queue length and waiting time results of ramp metering strategies

		7 th St.	SW Blvd.	18 th St.	67 th
ALINEA	Avg. Queue	48ft	413ft	195ft	N/A
	Max. Queue	238ft	1712ft	440ft	N/A
	Spillback %	0.0%	12.8%	2.1%	N/A
	Avg. Wait Time	6.6 sec	71.3 sec	38.5 sec	53.5 sec
	Max Wait Time	61.2	372.3 sec	82 sec	148 sec
HERO	Avg. Queue	99ft	430ft	160ft	N/A
	Max Queue	230ft	930ft	380ft	N/A
	Spillback %	0.0%	0.0%	0.05%	N/A
	Avg. Wait Time	18.6 sec	79.8 sec	31.6 sec	53 sec
	Max Wait Time	61.2 sec	200.8 sec	73.5 sec	155 sec

From Table 16 it can be seen that the average queue lengths and average waiting time at 7th St and SW Blvd were greater in ALINEA than in the HERO control scenario. However, at 18th St. the average queue length and average waiting time in HERO were lower compared to ALINEA. The maximum queue lengths at all the on-ramps were lower in HERO compared to ALINEA. The queue spillback percentage to the arterials at all the on-ramps were zero in the HERO control scenario. However, in ALINEA there were queue spillbacks at SW Blvd and 18th St. The average queue lengths at SW Blvd and 7th St on-ramp were longer in HERO because they act as slave ramps and create queues when the 18th St ramp (master) queue length exceed the queue activation threshold while in ALINEA the on-ramps do not coordinate and act isolated. ALINEA and HERO have similar waiting times on 67th St. because ALINEA was used in both control scenarios on 67th on-ramp. The queue lengths at 67th St on-ramp are not available because the queue detector was placed in a wrong position in the simulation model.

6.2. Throughput

Throughput is the volume of vehicles passing through a segment during a specific time intervals just before the breakdown event occurs. The three control scenarios were evaluated based on their throughput at the ramp meter locations. Given the large variability of demands in all scenarios generated in this research, throughput was measured only for the seed numbers that experience high, medium, and low demand volumes, chosen based on visually inspecting the speed heat maps. The per-lane throughput at the four ramp metered sites, as well as the change in throughput between the No Control and the metered scenarios, are presented in Table 17.

Table 17 Throughput and percentage change of throughput for the three control scenarios

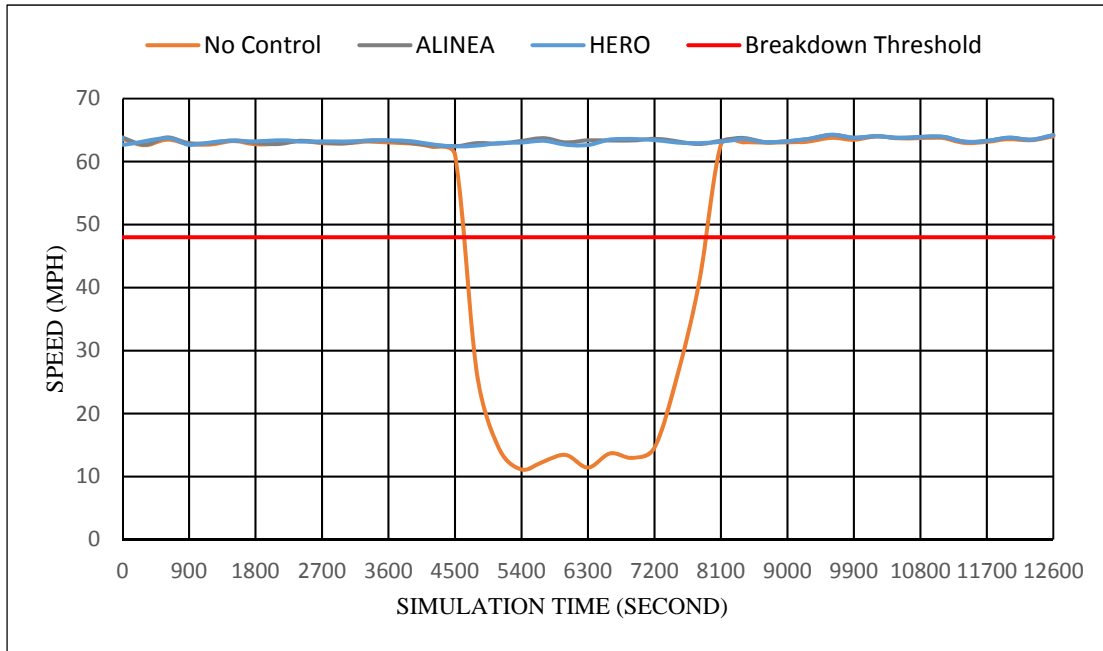
	7 th St.	SW Blvd.	18 th St.	67 th St.
High Demand Throughput (veh/h/ln)				
No Control	1536	1592	1648	1794
ALINEA	1604 (4.4%)	1624 (2.0%)	1724 (4.6%)	1776 (-1.0%)
HERO	1612 (4.9%)	1640 (3.0%)	1728 (4.9%)	1830 (2.0%)
Medium Demand Throughput (veh/h/ln)				
No Control	1600	1508	1648	2104
ALINEA	1600 (0%)	1632 (8.2%)	1696 (2.9%)	2180 (3.6%)
HERO	1600 (0%)	1700 (12.7%)	1660 (0.07%)	2160 (2.7%)
Low Demand Throughput (veh/h/ln)				
No Control	1600	1536	1596	2336
ALINEA	1600	1604 (4.4%)	1688 (5.8%)	2048 (-12.3%)
HERO	1600	1604 (4.4%)	1608 (0.8%)	2280 (-2.4%)

From Table 17 it can be seen that during a high, medium and low demand the throughput increased with the implementation of ramp metering strategies at the SW Blvd. and 18th St. junctions. At the 7th St. junction during the high demand scenario the throughput increased with

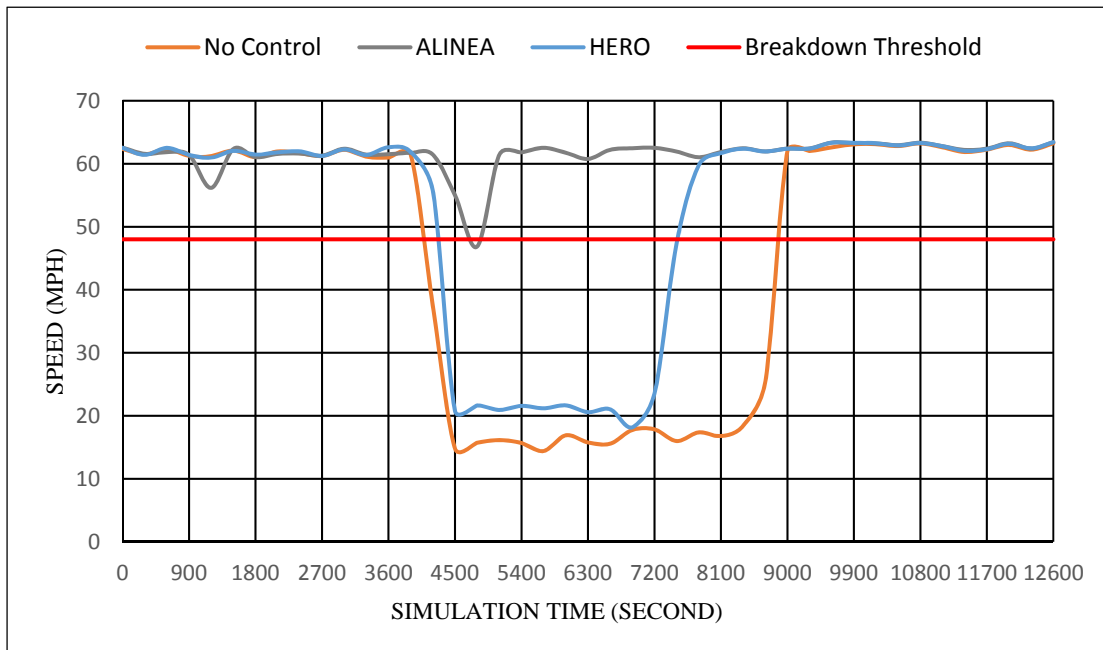
ramp metering, however, during medium and low demand conditions there was no change in throughput because it was operating under free flowing conditions in all three control scenarios. At the 67th St junction during high demand the throughput increased in HERO and decreased in ALINEA. The throughput decrease in ALINEA could be attributed to the fluctuations in vehicle demand arriving in the chosen 5-minute interval. At medium demand both algorithms resulted in higher throughput, and at low demand the throughput decreased in both ALINEA and HERO. The throughput decrease in HERO and ALINEA is attributed to the increase in traffic demand on 67th St. compared to the No Control scenario which resulted in congestion. At low demand, 67th St. exhibits little to no congestion in the No Control scenario. The SW Blvd. junction benefited the most from implementing ramp metering at all demand variations and 67th St. junction suffered with lower throughput during low demand.

6.3. Congestion Duration

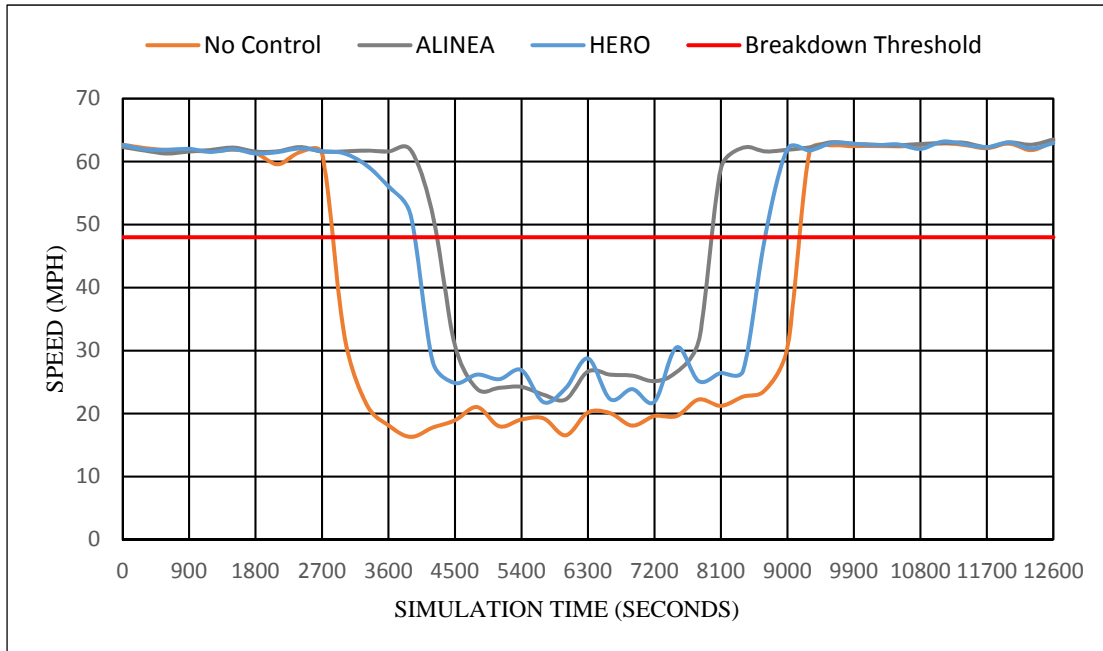
Congestion duration is the duration in which the mainline corridor speed is below 75% of the free flow speed (HCM6, 2016; Asgharzadeh and Kondyli, 2016) (i.e., experiences congestion). The free flow speed was considered to be 64 mph based on reviewing KC Scout Portal data. Hence, the speed at which the breakdown occurs is assumed 48 mph. An Analysis of breakdown events was conducted at the mainline upstream detectors where the ramp meters are installed. Typically, the downstream detectors should be used to analyze the breakdown; however, because in microsimulation the breakdown occurs before the end of the acceleration lane, the upstream detector was used for the analysis. The seed number with high demand used in the throughput analysis was used in measuring the congestion duration. Speed-time diagrams for the metered ramps are presented in Figure 20 (a-d). In addition, a summary of congestion duration at these locations is presented in **Error! Reference source not found.**



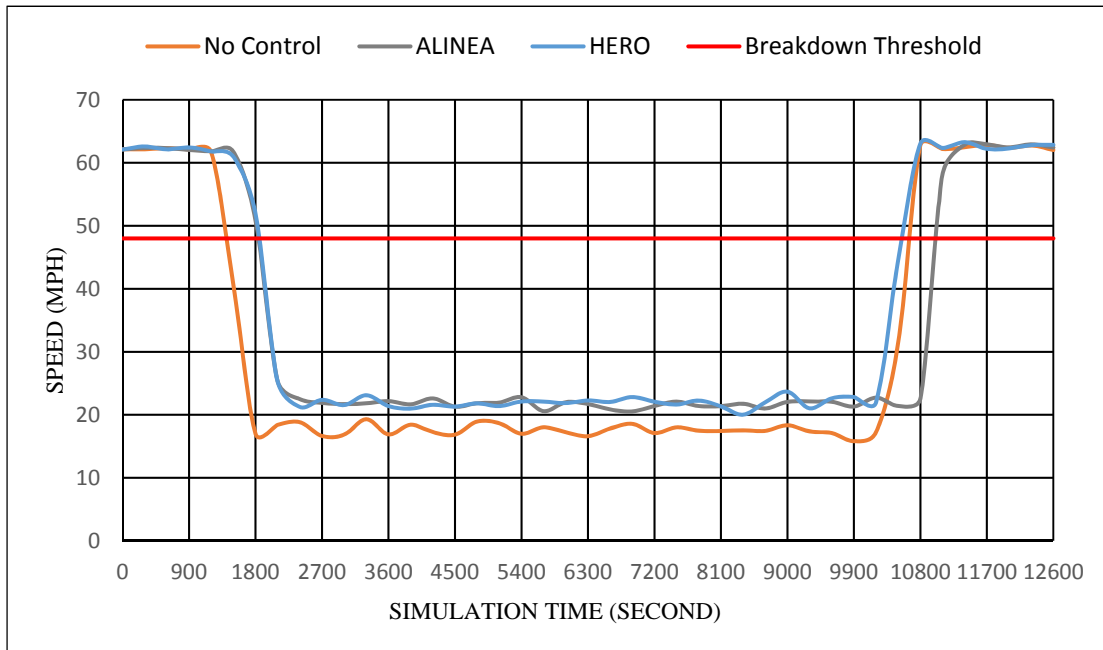
(a)



(b)



(c)



(d)

Figure 20 Speed-time graph indicating congestion at (a) 7th St. merge, (b) SW Blvd. upstream, (c) 18th St. upstream, and (d) 67th St. upstream detectors

Table 18 Congestion duration for the three control scenarios

	7 th St.	SW Blvd.	18 th St.	67 th S.
No Control	50-minutes	80-minutes	105-minutes	155-minutes
ALINEA	0 minutes	0 minutes	60-minutes	155-minutes
HERO	0 minutes	50-minutes	80-minutes	150-minutes

Based on Figure 21 (a-d) and Table 19, the implementation of ramp meters has positive effect on congestion duration. The 7th St, SW Blvd, and 18th St bottlenecks exhibited significant reduction in congestion duration. However, 67th St. did not experience significant change in congestion duration, because the mainline at the ramp location experiences higher demand incoming from the upstream facility with higher speeds. ALINEA also outperformed HERO as it resulted in reduced congestion duration at all the on-ramp locations except at 67th St. in which HERO was superior. Also, based on Figure 22 (a-d) it can be seen that implementing a ramp metering strategy improves speeds during breakdown compared to the No Control scenario.

6.4. Mean Speed

Speed is used as a performance measurement to evaluate the three control scenarios. Spot mean speed of upstream detectors at the on-ramp locations were averaged for each control scenario. The single seed number with high demand used in the throughput analysis was used to evaluate the spot mean speed. Only one seed day was used and not the average of all 240 demand scenarios because some seeds do not have any congestion, and therefore, they do not show the effectiveness of ramp metering. The spot mean speed and its percentage change in the three control scenarios are presented in Table 20.

Table 20 Spot mean speed and percentage change of speed for the three control scenarios

	7 th St.	SW Blvd.	18 th St.	67 th St.
No Control	51.5mph	45.8mph	41.9mph	30.9mph
ALINEA	63.3mph (23.0%)	61.5mph (34.3%)	51.8mph (23.6%)	33.8mph (9.4%)
HERO	63.3mph (23.0%)	52.1mph (13.8%)	48.6mph (16.0%)	35.4mph (14.6%)

It can be seen from Table 20 that during high demand the spot mean speed during the peak period improves with the implementation of ramp metering algorithms at all ramp metered locations. In addition, at SW Blvd. and 18th St junctions, the spot mean speed during the ALINEA scenario is higher compared to HERO. However, at 67th St junction the spot mean speed in HERO scenario is higher compared to ALINEA.

6.5. Results Discussion

Through simulation it was found that implementing a ramp metering strategy could improve or deteriorate the facility operations in some cases, depending on the location of the bottlenecks and the specifications of the ramp metering algorithm. In this research, several performance measurements were used to evaluate the pros and cons of each control scenario.

It was found that the travel time and travel time reliability for the entire facility did not exhibit significant improvement when a ramp metering strategy was implemented. However, the first half of the facility underwent drastic improvement in travel time and travel time reliability, while the second half of the facility experienced worse travel times and travel time reliability after implementing ramp metering strategies. The ramp meter on 67th created a brief good impact on travel times, but it was overshadowed with the excessive vehicle demand incoming from the first

half of the facility. In addition, it was also found that ALINEA had better travel times and travel time reliability compared to HERO.

Throughput was also used as a performance measurement to evaluate the three control scenarios. The results showed that the ramp metering algorithms improved the throughput on the mainline during heavy, medium, and low demand scenarios. However, in the case of low demand scenario, the 67th St bottleneck experienced lower throughput with the implementation of ramp metering. This occurs because in the No Control scenario, the 67th St bottleneck experiences little to no congestion duration while the north part bottlenecks experience breakdowns. After implementing the ramp metering algorithms, the bottlenecks in the north part of the freeway dissolve and the traffic flow incoming to 67th St bottleneck increases which creates longer congestion periods.

Implementing ramp metering strategies reduced the congestion duration at 7th St. merge, SW Blvd. upstream, and 18th St. upstream areas. However, at 67th St the congestion duration increased in the case of ALINEA and remained the same with HERO. ALINEA was more effective in reducing the congestion duration compared to HERO, at 7th St, SW Blvd, and 18th St areas.

Queue length and waiting time on the on-ramp during ramp metering were also used as performance measures. ALINEA uses a queue flush system, which discharges the queue at maximum metering rate when the on-ramp queue reaches 75% of the queue capacity. HERO uses a queue control system, which adjusts the metering rate, by setting the current queue not to exceed the queue capacity. It was found that the HERO algorithm prevented queue spillbacks on the adjacent arterials, whereas these spillbacks were not avoided in ALINEA.

6.6. Limitations

The study limitations are:

1. The calibration process was not successful at a few locations, as shown in Figure 12. Congestion at Antioch St was not replicated. This could be attributed to detector readings that were unrealistically high or low (see also Figure 14).
2. Travel times for the first and second half of I-35 are the summation of vehicle travel times in each segment at the same time interval and not the travel time for vehicles travelling the entire section continuously.
3. The queue detector at 67th St was placed in the wrong location; hence, queue data for that on-ramp were not available.
4. The arterials connecting to the on-ramps are signalized intersections. The demands and signal timing for the arterials are not available, so the signalized intersections were not modeled in VISSIM.
5. Truck percentages were not available in the KC Scout data, so these were assumed as given from KDOT VISSIM model (3.75%).

7. CONCLUSION AND RECOMMENDATIONS

This thesis evaluated the ALINEA and HERO ramp metering algorithms on I-35 southbound corridor through simulation, and compared them with the No Control scenario. The evaluation was conducted using the VISSIM microsimulation software using data obtained from KC Scout Portal during the peak periods, for one year (April 1, 2016 to March 31, 2017). Days with adverse weather conditions or incidents were discarded as the focus was on the effect of ramp metering on recurring traffic operations.

The I-35 southbound corridor under study runs from Cambridge Circle junction to 75th Street interchange. The ramp meters are installed on 7th street, Southwest Blvd, 18th street, and 67th street on-ramps. The corridor was modeled and calibrated in VISSIM using a day that traffic incidents, bad weather, and bad detector data did not occur. Then, traffic data for 60 demand scenarios were generated. Each demand scenario was run in simulation four times with different seed numbers resulting in a total of 240 seed days.

No control, ALINEA, and HERO are the three control scenarios evaluated through microsimulation. The control scenarios were evaluated using the same 60 demand scenarios and seed numbers. The performance measures used for the evaluation are travel time, travel time reliability, congestion duration, mean speed, queue length, and queue waiting time. The results indicate that ALINEA and HERO have slightly lower travel times than the No Control scenario for the entire facility. However, if the facility is split halfway (at Metcalf avenue off-ramp), the first half (Cambridge Circle to Metcalf Avenue off-ramp) results show that ALINEA and HERO resulted in lower travel times than the No Control scenario by a significant margin. The results of the second half of the facility (Metcalf Avenue to 75th Street) show that the travel time increased when ALINEA and HERO ramp metering algorithms were implemented. This suggests that the

first half bottlenecks experience shorter breakdowns and the traffic flows to the second half bottleneck faster. This phenomena worsens the traffic conditions at the downstream bottleneck.

Comparing the two ramp metering algorithms, ALINEA generated better travel times than HERO, however, the difference is small. Travel time reliability was measured through the cumulative probability of travel times, travel time index, buffer time, and buffer index. For the entire facility the travel time reliability was slightly better in ALINEA and HERO scenarios than the No Control scenario. ALINEA resulted in lower congestion duration than HERO or the No Control scenario at 7th St merge, SW Blvd, and 18th St locations. At 67th St, there was no change in the congestion duration between the three control scenarios. The spot mean speed of the peak period and the speed during breakdown at the ramp meter locations are higher in ALINEA and HERO scenarios than the No Control scenario. Queue length and queue wait time are an indication if the ramp metering algorithm is serving the queue fairly without affecting the arterials. Simulation results show that HERO was more effective in managing queues than ALINEA by minimizing the spillback to the arterials to zero while maximizing the usage of the ramp storage. HERO is used for on-ramps that are located next to each other, while ALINEA is used for an isolated on-ramp. In addition, ALINEA could be better than HERO if sufficient ramp queue length is available or if the arterials could handle spillbacks.

Previous research suggests that the implementation of ramp metering strategies improve the traffic conditions on the mainline. This was also observed in this thesis as ALINEA and HERO improved the traffic conditions at several locations. The simulation results of the three control scenarios suggest that if the traffic conditions at the north part of the freeway is important, then implementing the ramp metering strategies prove to be significantly beneficial to the traffic conditions. ALINEA provides better mainline traffic conditions compared to HERO; however, it

creates spillbacks to the arterials from the on-ramp while HERO does not. The fact that traffic operations worsened on the second half of the facility, and more specifically at the 67th St bottleneck, suggests that ramp metering is not effective at this location, and that perhaps another freeway management should be evaluated.

Conducting an analysis using microsimulation prior to field implementation, helps to reveal the strengths and weaknesses of the strategy, and the areas for improvement. Also, it gives the decision makers information that helps them decide if they want to go with a strategy or abort it.

The following recommendations are offered based on the findings of this research:

- Based on the performance measures results, ALINEA is superior compared to HERO in improving the mainline traffic conditions. However, the ramps suffered longer queues in ALINEA compared to HERO and also had spillbacks to the arterials while HERO did not create any spillbacks. A balance between serving the mainline and on-ramps is better than serving only the mainline while deteriorating the arterials. Thus, HERO is more efficient than ALINEA.
- This thesis covered ALINEA with a queue flush system that assigns maximum metering rate when the ramp queue reaches a certain threshold. Another approach in this study could be done by coding ALINEA to a use queue control approach similar to the one in HERO instead of the flush system.
- This thesis evaluates the effects of ALINEA and HERO on four junctions along I-35 southbound. It would be useful to test if ramp metering would be more effective in relieving congestion at 67th St, if applied to additional junctions as well.

- HERO uses a combination of master ramp queue activation percentage and slave ramp queue creation percentage. This research used 50% for both the queue activation and creation. Other combinations could be used as they may improve the conditions on the on-ramp and the mainline.
- The ramp meters in this research allowed one vehicle per green. Future research could be conducted by adjusting the ramp meters to allow two vehicles per green.

8. REFERENCES

- Armstrong, J. (2011). "Optimizing the Efficiency and Equity of Traffic Flow" (Doctoral dissertation, Carleton University Ottawa, Canada).
- Asgharzadeh, M., & Kondyli, A. (2018). "Comparison of Highway Capacity Estimation Methods". Transportation Research Board Annual Meeting, Washington, DC.
- Bhourri, N., Haj-Salem, & H., Kauppila, (2013). "Isolated Versus Coordinated Ramp Metering: Field Evaluation Results of Travel Time Reliability and Traffic Impact". In Transportation Research Part C: Emerging Technologies 28 (2013), pp.155-167.
- Bogenberger, K., & May, A. D. (1999). "Advanced Coordinated Traffic Responsive Ramp Metering Strategies". UCB-ITS-PWP-99-1. Institute of Transportation Studies, University of California, Berkeley, Calif.
- Chu, L., Liu X., Recker, W., Zhang, H.M. (2004). "Performance Evaluating of Adaptive Ramp Metering Algorithms". ASCE Journal of Transportation Engineering, Vol. 130 (3), pp. 330-338.
- Faulkner, L., Dekker, F., Gyles, D., Papamichail, I., & Papageorgiou, M. (2014). "Evaluation of HERO-Coordinated Ramp Metering Installation at M1 and M3 Freeways in Queensland, Australia". Transportation Research Record: Journal of the Transportation Research Board, (2470), pp. 13-23.
- Greenshields, B. D., Ws Channing, and Hh Miller, (1935). "A Study of Traffic Capacity". Highway Research Board Proceedings. Vol. 1935. National Research Council (USA), Highway Research Board.

- Jacobson, L., Stribiak, J., Nelson, L., & Sallman, D. (2006). "Ramp Management and Control Handbook". (No. FHWA-HOP-06-001). United States. Federal Highway Administration.
- Koble, H. M. and Samant, V. S. (1980). "Control Strategies in Response to Freeway Incidents". (Vol. 4). Federal Highway Administration, Offices of Research & Development, Traffic Systems Division.
- Kwon, E., Nanduri, S., Lau, R., & Aswegan, J. (2001). "Comparative Analysis of Operational Algorithms for Coordinated Ramp Metering". Transportation Research Record: Journal of the Transportation Research Board, (1748), pp. 144-152.
- Levinson, D. and L. Zhang (2006). "Ramp Meters on Trial: Evidence from the Twin Cities Metering Holiday". Transportation Research Part A: Policy and Practice 40(10): pp. 810-828.
- Leyn, U. and P. Vortisch (2015). "Calibrating Vissim for the German Highway Capacity Manual". Transportation Research Record: Journal of the Transportation Research Board (2483): pp. 74-79.
- Masher, D. P., Ross, D. W., Wong, P. J., Tuan, P. L., Zeidler, H. M., & Petracek, S. (1975). "Guidelines for Design and Operation of Ramp Control Systems" Stanford Research Institute, Menlo Park, Calif.
- Ozbay, K., Yasar, I., & Kachroo, P. (2004). "Comprehensive Evaluation of Feedback-Based Freeway Ramp Metering Strategy by Using Microscopic Simulation: Taking Ramp Queues into Account". Transportation Research Record: Journal of the Transportation Research Board, (1867), pp. 89-96.

- Paesani et al., (1997). "System Wide Adaptive Ramp Metering (SWARM). Merging the Transportation and Communications Revolutions". Abstracts for ITS America Seventh Annual Meeting and Exposition.
- Papageorgiou, M., Blosseville, J. M., & Haj-Salem, H. (1990). "Modelling and Real-time Control of Traffic Flow on the Southern Part of Boulevard Périphérique in Paris: Part II: Coordinated On-ramp Metering". *Transportation Research Part A: General*, 24(5), pp. 361-370.
- Papageorgiou, M., Hadj-Salem, H., & Blosseville, J. M. (1991). "ALINEA: A Local Feedback Control Law for On-ramp Metering". *Transportation Research Record*, 1320(1), pp. 58-67.
- Papageorgiou, M., Hadj-Salem, H., & Middelham, F. (1997). "ALINEA Local Ramp Metering: Summary of Field Results". *Transportation Research Record: Journal of the Transportation Research Board*, (1603), pp. 90-98.
- Papageorgiou, M., Kosmatopoulos, E., Papamichail, I., & Wang, Y. (2007). "ALINEA Maximizes Motorway Throughput-an Answer to Flawed Criticism". *Traffic Engineering and Control*, 48(6), pp. 271-276.
- Papamichail, I., Papageorgiou, M., Vong, V., & Gaffney, J. (2010). "Heuristic Ramp Metering Coordination Strategy Implemented at Monash Freeway, Australia". *Transportation Research Record: Journal of the Transportation Research Board*, (2178), pp. 10-20.
- PTV Group. (2016). "PTV Vissim 9.0 User Manual". PTV AG, Karlsruhe, Germany.
- Transportation Research Board, Highway Capacity Manual 6th Edition: A Guide for Multimodal Mobility Analysis, Washington, DC, 2016.

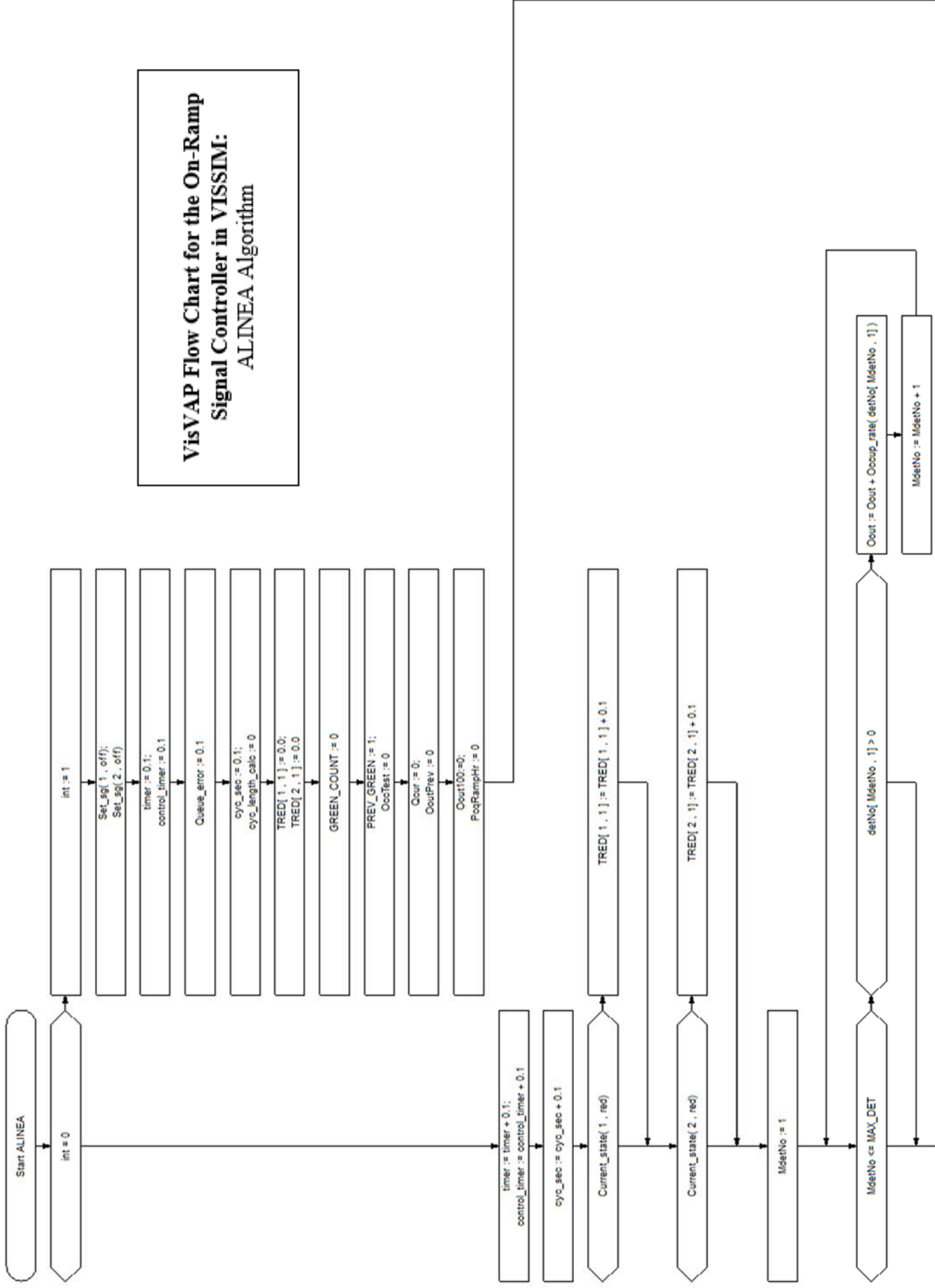
- Shaaban, K., Khan, M. A., & Hamila, R. (2016). "Literature Review of Advancements in Adaptive Ramp Metering". *Procedia Computer Science*, 83, pp. 203-211.
- Smaragdis, E. and M. Papageorgiou (2003). "Series of New Local Ramp Metering Strategies". *Transportation Research Record: Journal of the Transportation Research Board* (1856): pp. 74-86.
- Stephanedes, Y. J. (1994). "Implementation of on-line Zone Control Strategies for optimal ramp metering in the Minneapolis Ring Road" *Proc. 7th Int. Conf. Road Traffic Monitoring Control*, pp. 181-184.
- Taale, H., Slager, J., & Rosloot, J. (1996). "The Assessment of Ramp Metering Based on Fuzzy Logic". In *3rd ITS World Congress in Orlando*.
- Taylor, C., Meldrum, D., & Jacobson, L. (1998). "Fuzzy Ramp Metering: Design Overview and Simulation Results". *Transportation Research Record: Journal of the Transportation Research Board*, (1634), pp. 10-18.
- Taylor, C. E. and D. R. Meldrum (2000). "Evaluation of a Fuzzy Logic ramp metering Algorithm: a Comparative Study among Three Ramp Metering Algorithms Used in the Greater Seattle Area", *Washington State Department of Transportation*.
- Wang, Y., Kosmatopoulos, E. B., Papageorgiou, M., & Papamichail, I. (2014). "Local Ramp Metering in the Presence of a Distant Downstream Bottleneck: Theoretical Analysis and Simulation study". *IEEE Transactions on Intelligent Transportation Systems*, 15(5), pp. 2024-2039.
- Zadeh, L. A. (1965). "Fuzzy sets." *Information and Control* 8(3): pp. 338-353.

Zaidi, K., Milojevic, M. B., Rakocevic, V., Nallanathan, A., & Rajarajan, M. (2016). "Host-based Intrusion Detection for VANETs: a Statistical Approach to Rogue Node Detection". IEEE transactions on vehicular technology, 65(8), pp. 6703-6714.

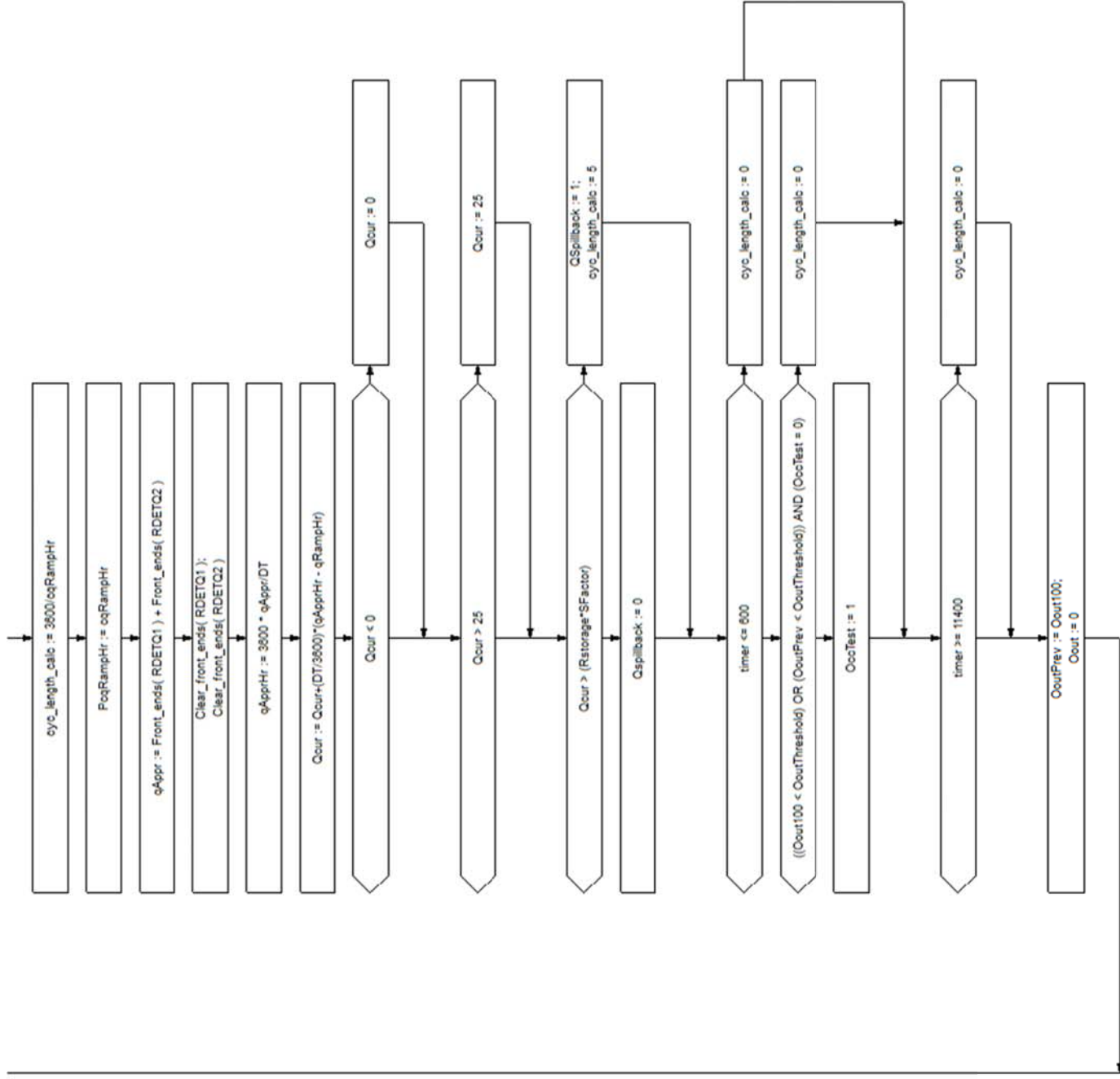
Zhang, M., Kim, T., Nie, X., Jin, W., Chu, L., & Recker, W. (2001). "Evaluation of On-ramp Control Algorithms" California PATH Research Report UCB-ITS-PRR-2001-36. Berkeley, Calif.

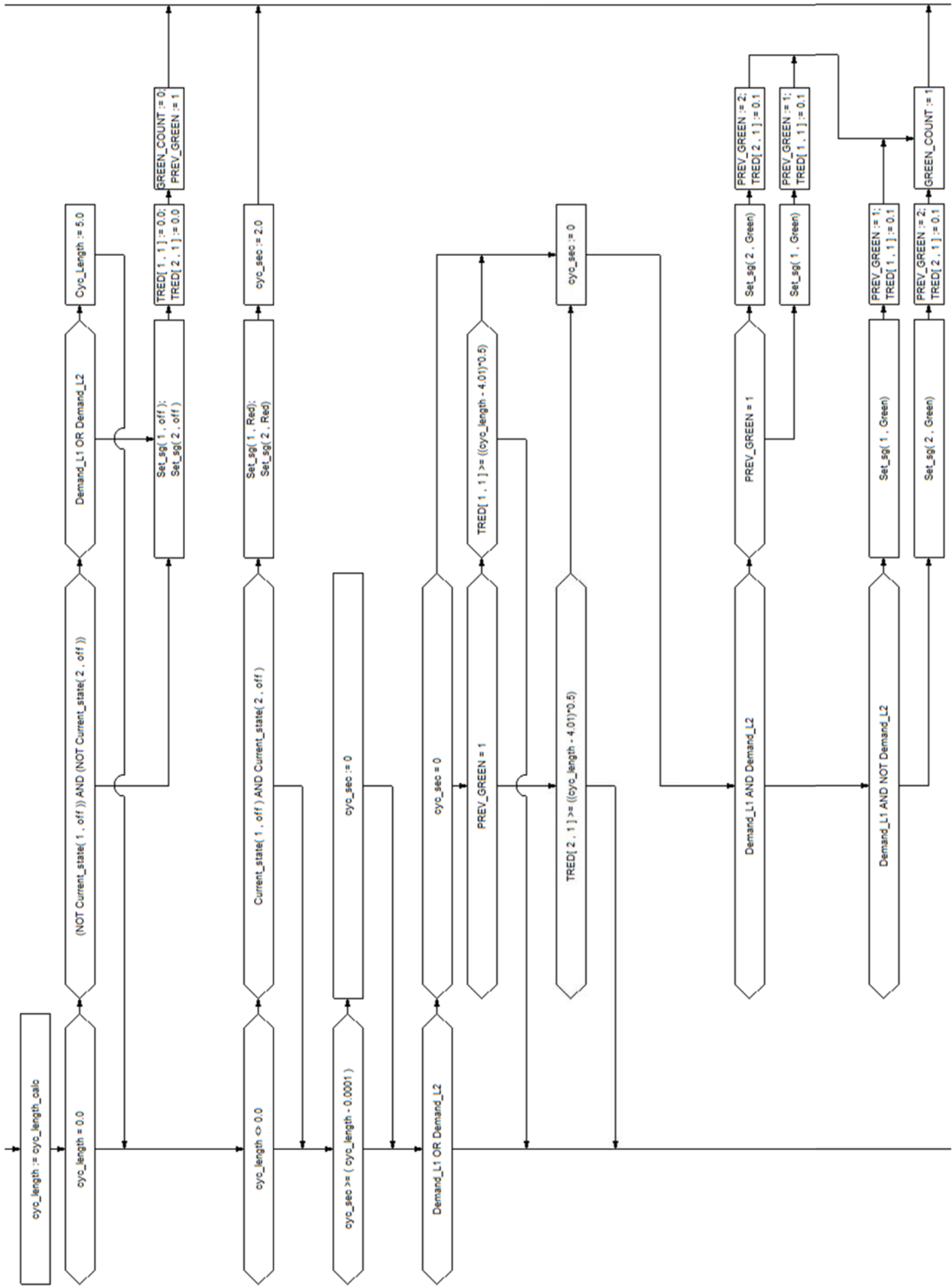
APPENDIX A

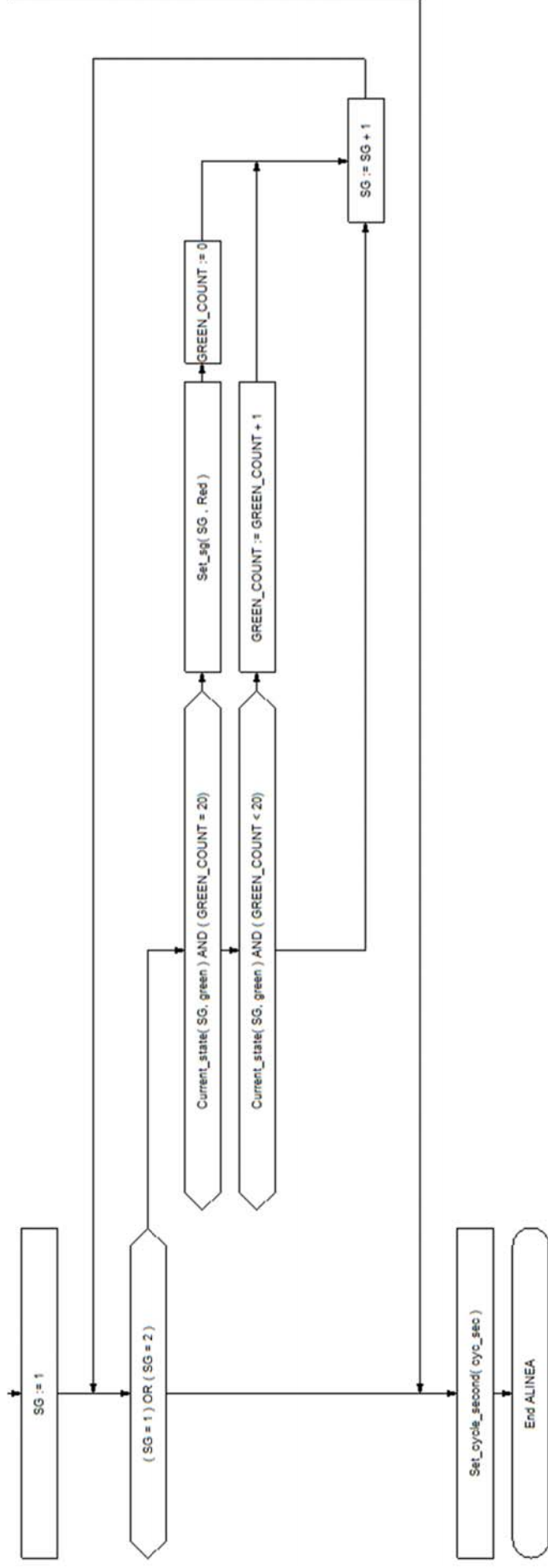
ALINEA Ramp Metering Algorithm Code in VISSIM











PARAMETERS, ARRAYS and EXPRESSIONS USED IN THE ALINEA ALGORITHM VisVAP FLOW CHART

PARAMETERS	Gen	Prog 1	Prog 2,...8	Comment
RDETAPP_L1	10			Ramp detector approaching stop bar, lane 1.
RDETAPP_L2	11			Ramp detector approaching stop bar, lane 2.
RDETDEP_L1	12			Ramp detector departing stop bar, lane 1.
RDETDEP_L2	13			Ramp detector departing stop bar, lane 2.
RDETDEP_Ler1	14			Ramp detector departing stop bar lane 1 (Queue error correction).
RDETDEP_Ler2	15			Ramp detector departing stop bar lane 2 (Queue error correction).
RDETQ1	18			Ramp queue detector lane 1.
RDETQ2	19			Ramp queue detector lane 2.
RDETQer1	16			Ramp queue detector lane 1 (Queue error correction).
RDETQer2	17			Ramp queue detector lane 2 (Queue error correction).
MAX_DET	3			Number of mainline detectors.
DT	20			Ramp metering control time step.
KR	70			Regulator parameter.
OCC_OPT	13			Desired occupancy.
OoutThreshold	11			Mainline occupancy threshold to activate ramp metering.
Rstorage	24			Ramp storage (vehicles).
Sfactor	0.75			Safety factor for ramp storage.
MaxcqRampHr	720			Maximum ramp metering flow rate per lane.
MincqRampHr	200			Minimum ramp metering flow rate per lane.

ARRAYS	Dim2	Dim1	[1]	Comment
TRED	2	1	0	Time since signal 1 turned red.
TRED[2]			0	Time since signal 2 turned red.
DetNo	3	1	101	1 st mainline detector
detNo[2]			102	2 nd mainline detector
detNo[3]			103	3 rd mainline detector

EXPRESSIONS	Contents	Comment
Demand_L1	Detection(RDETAPP_L1)	Detects vehicles waiting at the stop bar in Lane 1
Demand_L2	Detection(RDETAPP_L2)	Detects vehicles waiting at the stop bar in Lane 2

```

                                ALINEA
PROGRAM ALINEA_7th; /*
VAP_FREQUENCY 10;

CONST
    RDETAPP_L1 = 10,
    RDETAPP_L2 = 11,
    RDETDEP_L1 = 12,
    RDETDEP_L2 = 13,
    RDETDEP_Ler1 = 14,
    RDETDEP_Ler2 = 15,
    RDETQ1 = 18,
    RDETQ2 = 19,
    RDETQer1 = 16,
    RDETQer2 = 17,
    MAX_DET = 3,
    DT = 20,
    KR = 70,
    OCC_OPT = 13,
    OoutThreshold = 11,
    RStorage = 24,
    SFactor = 0.75,
    MaxcqRampHr = 720,
    MincqRampHr = 200;

/* ARRAYS */
ARRAY
    TRED[ 2, 1 ] = [[0], [0]],
    detNo[ 3, 1 ] = [[101], [102], [103]];

/* SUBROUTINES */

/* PARAMETERS DEPENDENT ON SCJ-PROGRAM */

/* EXPRESSIONS */
    Demand_L1 := Detection( RDETAPP_L1 );
    Demand_L2 := Detection( RDETAPP_L2 );

/* MAIN PROGRAM */

S00Z001:  IF int = 0 THEN
S01Z001:      int := 1;
S01Z002:      Set_sg( 1 , off); Set_sg( 2 , off);
S01Z003:      timer := 0.1; control_timer := 0.1;
S01Z004:      Queue_error := 0.1;
S01Z005:      cyc_sec := 0.1; cyc_length_calc := 0;
S01Z006:      TRED[ 1 , 1 ] := 0.0; TRED[ 2 , 1 ] := 0.0;
S01Z007:      GREEN_COUNT := 0;
S01Z008:      PREV_GREEN := 1; OccTest := 0;
S01Z009:      Qcur := 0; OoutPrev := 0;
S01Z010:      Oout100:=0; PcqRampHr := 0
ELSE
S00Z011:      timer := timer + 0.1; control_timer := control_timer + 0.1;
S00Z012:      cyc_sec := cyc_sec + 0.1;
S00Z013:      IF Current_state( 1 , red) THEN
S01Z013:          TRED[ 1 , 1 ] := TRED[ 1 , 1 ] + 0.1;
S00Z015:      IF Current_state( 2 , red) THEN
S01Z015:          TRED[ 2 , 1 ] := TRED[ 2 , 1 ] + 0.1;
S00Z017:          MdetNo := 1;
S00Z019:          IF MdetNo <= MAX_DET THEN
S01Z019:              IF detNo[ MdetNo , 1 ] > 0 THEN
S02Z019:                  Oout := Oout + Occup_rate( detNo[ MdetNo , 1 ] );
S02Z020:                  MdetNo := MdetNo + 1;
GOTO S00Z019

```

```

ELSE
S00Z021:   IF Detection( RDETQer1 ) OR Detection( RDETQer2 ) THEN
S01Z021:     Queue_error := 0;
S00Z024:     IF Queue_error >=10.99 THEN
S01Z024:       Queue_error := 0;
S01Z025:       Qerror_exit := (Front_ends( RDETDEP_Ler1 )) + (Front_ends(
RDETDEP_Ler2 ));
S01Z026:       Clear_front_ends( RDETDEP_Ler1 ); Clear_front_ends(
RDETDEP_Ler2 );
S01Z027:       Qerror_enter := Front_ends( RDETQer1 ) + Front_ends(
RDETQer2 );
S01Z028:       Clear_front_ends( RDETQer1 ); Clear_front_ends( RDETQer2
);
S01Z029:     IF (Qerror_exit <= 0) AND (Qerror_enter <= 0) THEN
S02Z029:       Qcur := 0;
S00Z031:       IF control_timer >= ( DT - 0.0001 ) THEN
S01Z031:         control_timer := 0;
S01Z032:         qRamp := (Front_ends( RDETDEP_L1 )) + (Front_ends(
RDETDEP_L2 ));
S01Z033:         Clear_front_ends( RDETDEP_L1 ); Clear_front_ends(
RDETDEP_L2 );
S01Z034:         qRampHr:= 3600*qRamp/DT;
S01Z035:         Oout:= Oout/MAX_DET/(DT*10);
S01Z036:         Oout100 := Oout*100;
S01Z037:         cqRampHr := PcqRampHr + KR*(OCC_OPT - Oout100);
S01Z038:         IF cqRampHr >= MaxcqRampHr THEN
S02Z038:           cqRampHr := MaxcqRampHr;
S01Z040:           IF cqRampHr <= MincqRampHr THEN
S02Z040:             cqRampHr := MincqRampHr;
S01Z042:             cyc_length_calc := 3600/cqRampHr;
S01Z043:             PcqRampHr := cqRampHr;
S01Z044:             qAppr := Front_ends( RDETQ1 ) + Front_ends( RDETQ2
);
S01Z045:             Clear_front_ends( RDETQ1 ); Clear_front_ends(
RDETQ2 );
S01Z046:             qApprHr := 3600 * qAppr/DT;
S01Z047:             Qcur := Qcur+(DT/3600)*(qApprHr - qRampHr);
S01Z048:             IF Qcur < 0 THEN
S02Z048:               Qcur := 0;
S01Z050:               IF Qcur > 25 THEN
S02Z050:                 Qcur := 25;
S01Z052:                 IF Qcur > (Rstorage*SFactor) THEN
S02Z052:                   QSpillback := 1; cyc_length_calc := 5;
S01Z055:                   IF timer <= 600 THEN
S02Z055:                     cyc_length_calc := 0;
S01Z059:                     IF timer >= 11400 THEN
S02Z059:                       cyc_length_calc := 0;
S01Z061:                       OoutPrev := Oout100; Oout := 0;
S00Z063:                       cyc_length := cyc_length_calc;
S00Z064:                       IF cyc_length = 0.0 THEN
S01Z064:                         IF (NOT Current_state( 1 , off )) AND
(NOT Current_state( 2 , off )) THEN
S02Z064:                           IF Demand_L1 OR Demand_L2 THEN
S03Z064:                             Cyc_Length := 5.0;
S00Z068:                             IF cyc_length <> 0.0 THEN
S01Z068:                               IF Current_state( 1 , off ) AND
Current_state( 2 , off ) THEN
S02Z068:                                 Set_sg( 1 , Red); Set_sg( 2 ,
Red);
S03Z068:                                 cyc_sec := 2.0
ELSE
S00Z070:                                   IF cyc_sec >= ( cyc_length -
0.0001 ) THEN

```


S01Z070:	cyc_sec := 0;
S00Z072:	IF Demand_L1 OR Demand_L2
THEN	
S01Z072:	IF cyc_sec = 0 THEN
S03Z075:	cyc_sec := 0;
S01Z079:	IF Demand_L1 AND
Demand_L2 THEN	
S02Z079:	IF PREV_GREEN = 1 THEN
S03Z079:	Set_sg(2 , Green);
S04Z079:	PREV_GREEN := 2;
TRED[2 , 1] := 0.1;	
S04Z083:	GREEN_COUNT := 1
	ELSE
S03Z080:	Set_sg(1 , Green);
S04Z080:	PREV_GREEN := 1;
TRED[1 , 1] := 0.1;	
	GOTO S04Z083
	END
	ELSE
S01Z082:	IF Demand_L1 AND NOT
Demand_L2 THEN	
S02Z082:	Set_sg(1 , Green);
S03Z082:	PREV_GREEN := 1;
TRED[1 , 1] := 0.1;	
	GOTO S04Z083
	ELSE
S02Z083:	Set_sg(2 , Green);
S03Z083:	PREV_GREEN := 2;
TRED[2 , 1] := 0.1;	
	GOTO S04Z083
	END
	END
	ELSE
S01Z073:	IF PREV_GREEN = 1 THEN
S02Z073:	IF TRED[1 , 1] >=
((cyc_length - 4.01)*0.5) THEN	
	GOTO S03Z075
	ELSE
S00Z084:	SG := 1;
S00Z086:	IF (SG = 1) OR (
SG = 2) THEN	
S01Z087:	IF Current_state(
SG, green) AND (GREEN_COUNT = 20) THEN	
S02Z087:	Set_sg(SG , Red
);	
S03Z087:	GREEN_COUNT :=
0;	
S04Z090:	SG := SG + 1;
	GOTO S00Z086
	ELSE
	IF
	GREEN_COUNT :=
	GOTO S04Z090
	ELSE
	GOTO S04Z090
	END
	END
	END
	END
	ELSE
S01Z075:	IF TRED[2 , 1] >=

```

((cyc_length - 4.01)*0.5) THEN
                                                    GOTO S03Z075
                                                    ELSE
                                                    GOTO S00Z084
                                                    END
                                                    END
                                                    END
                                                    ELSE
                                                    GOTO S00Z084
                                                    END
                                                    ELSE
                                                    GOTO S00Z072
                                                    END
                                                    END
                                                    END
                                                    GOTO S00Z070
                                                    END
S02Z066:                                     ELSE
Set_sg( 1 , off ); Set_sg( 2 , off
);
S03Z066:                                     TRED[ 1 , 1 ] := 0.0; TRED[ 2 , 1
] := 0.0;
S04Z066:                                     GREEN_COUNT := 0; PREV_GREEN := 1
                                                    END
                                                    ELSE
                                                    GOTO S02Z066
                                                    END
                                                    ELSE
                                                    GOTO S00Z068
                                                    END
                                                    ELSE
                                                    GOTO S01Z061
                                                    END
                                                    ELSE
S01Z056:                                     IF ((Oout100 < OoutThreshold) OR (OoutPrev
< OoutThreshold)) AND (OccTest = 0) THEN
S02Z056:                                     cyc_length_calc := 0;
                                                    GOTO S01Z059
                                                    ELSE
S01Z057:                                     OccTest := 1;
                                                    GOTO S01Z059
                                                    END
                                                    END
                                                    ELSE
S01Z053:                                     Qspillback := 0;
                                                    GOTO S01Z055
                                                    END
                                                    ELSE
                                                    GOTO S01Z052
                                                    END
                                                    ELSE
                                                    GOTO S01Z050
                                                    END
                                                    ELSE
                                                    GOTO S01Z042
                                                    END
                                                    ELSE
                                                    GOTO S01Z040
                                                    END
                                                    ELSE
                                                    GOTO S00Z063
                                                    END
                                                    ELSE

```

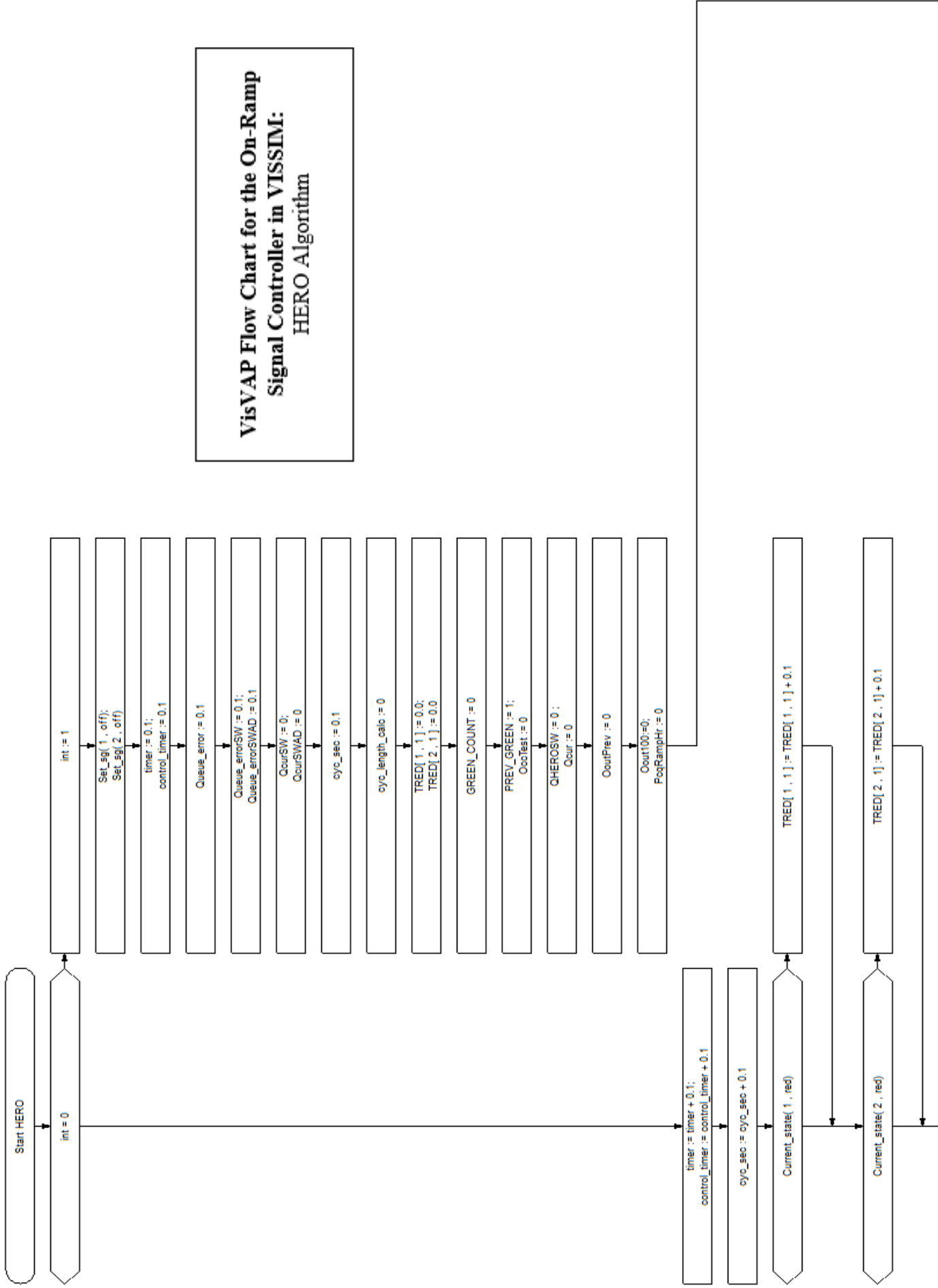
```

                                GOTO S00Z031
                                END
                                ELSE
                                GOTO S00Z031
                                END
                                ELSE
S00Z022: Queue_error := Queue_error + 0.1;
                                GOTO S00Z024
                                END
                                END
                                ELSE
                                GOTO S00Z021
                                END
                                ELSE
                                GOTO S00Z017
                                END
                                ELSE
                                GOTO S00Z015
                                END
                                END;
S00Z092: Set_cycle_second( cyc_sec )
PROG_ENDE: .
/*-----*/

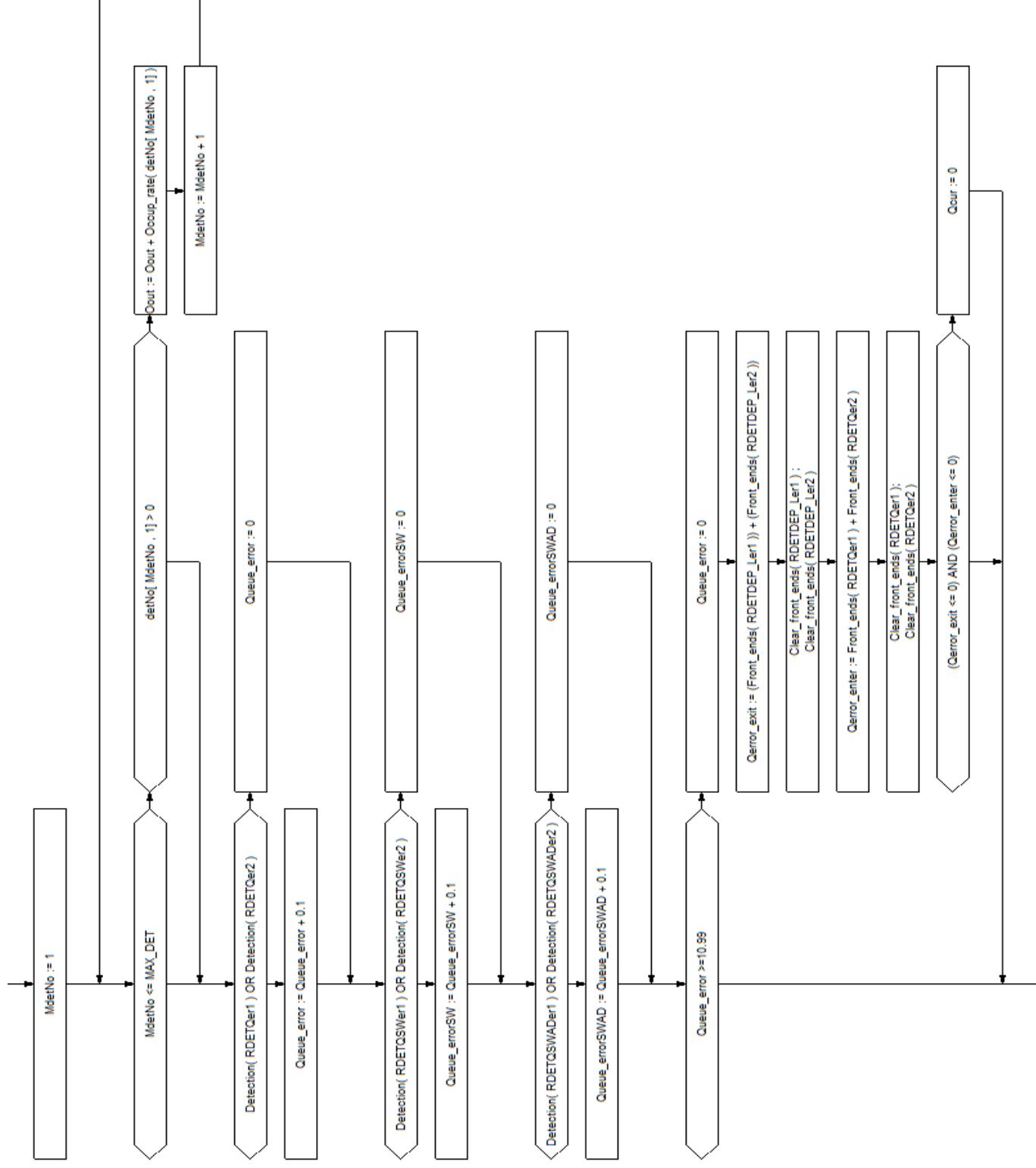
```

APPENDIX B

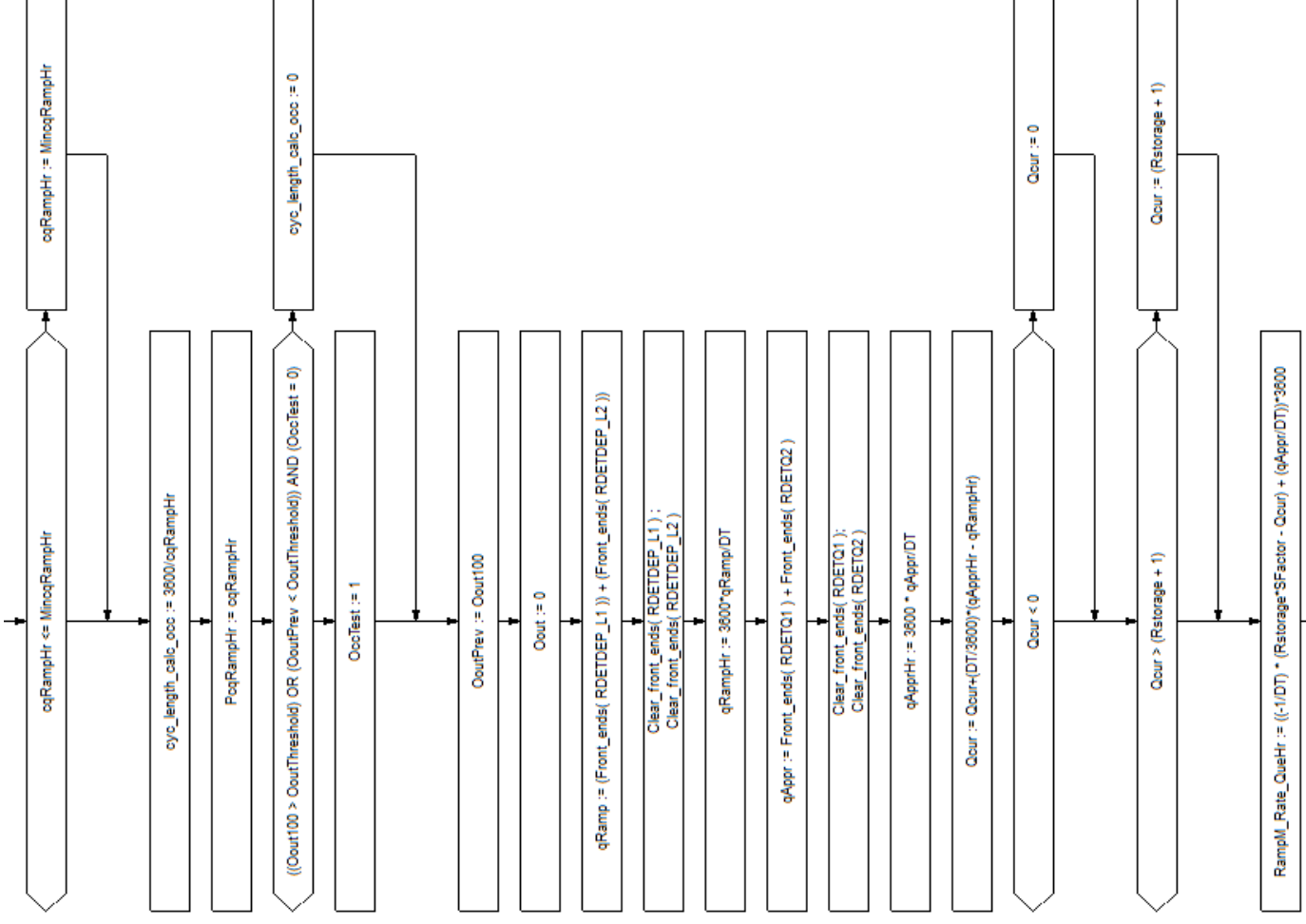
HERO Ramp Metering Algorithm Code in VISSIM

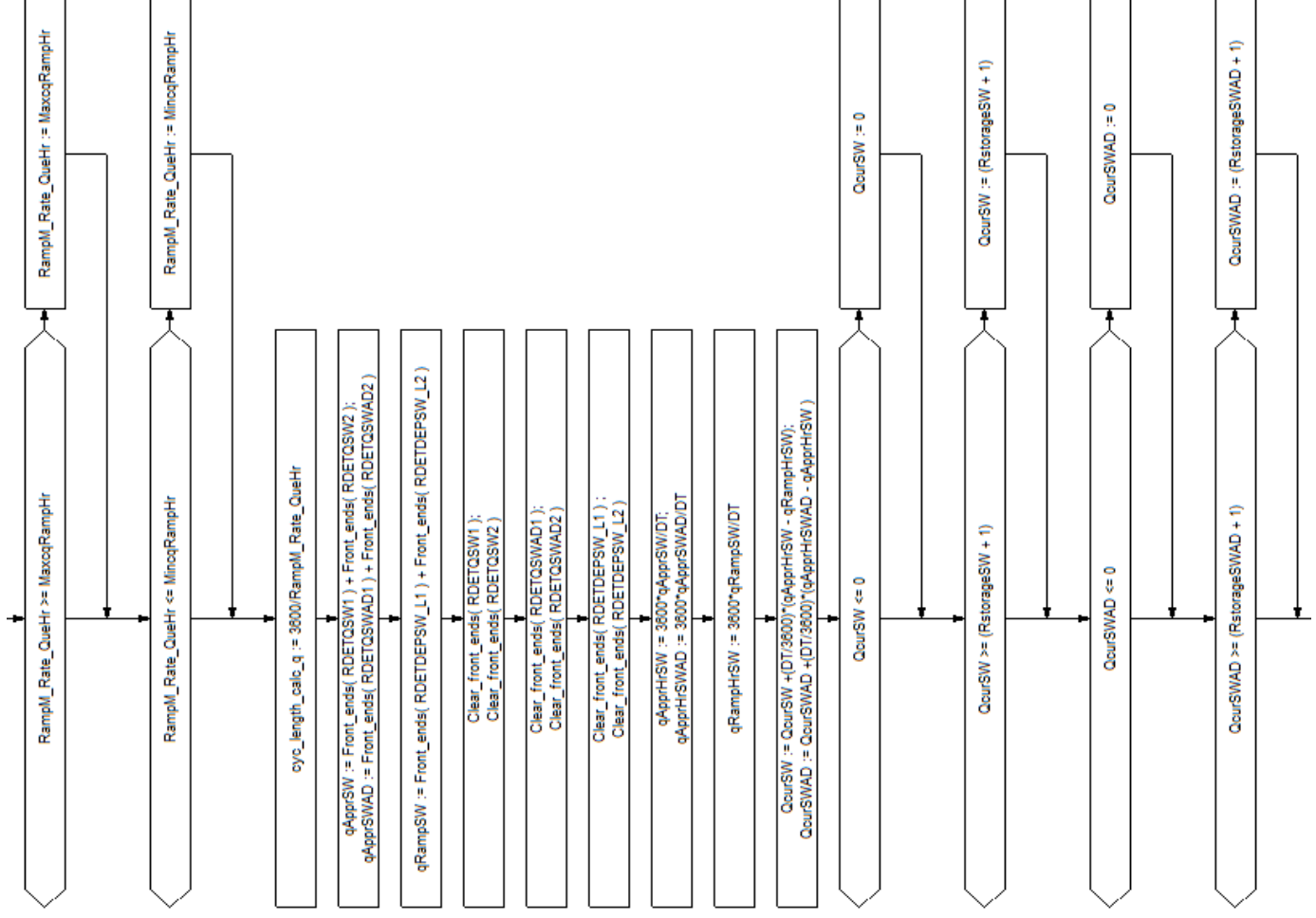


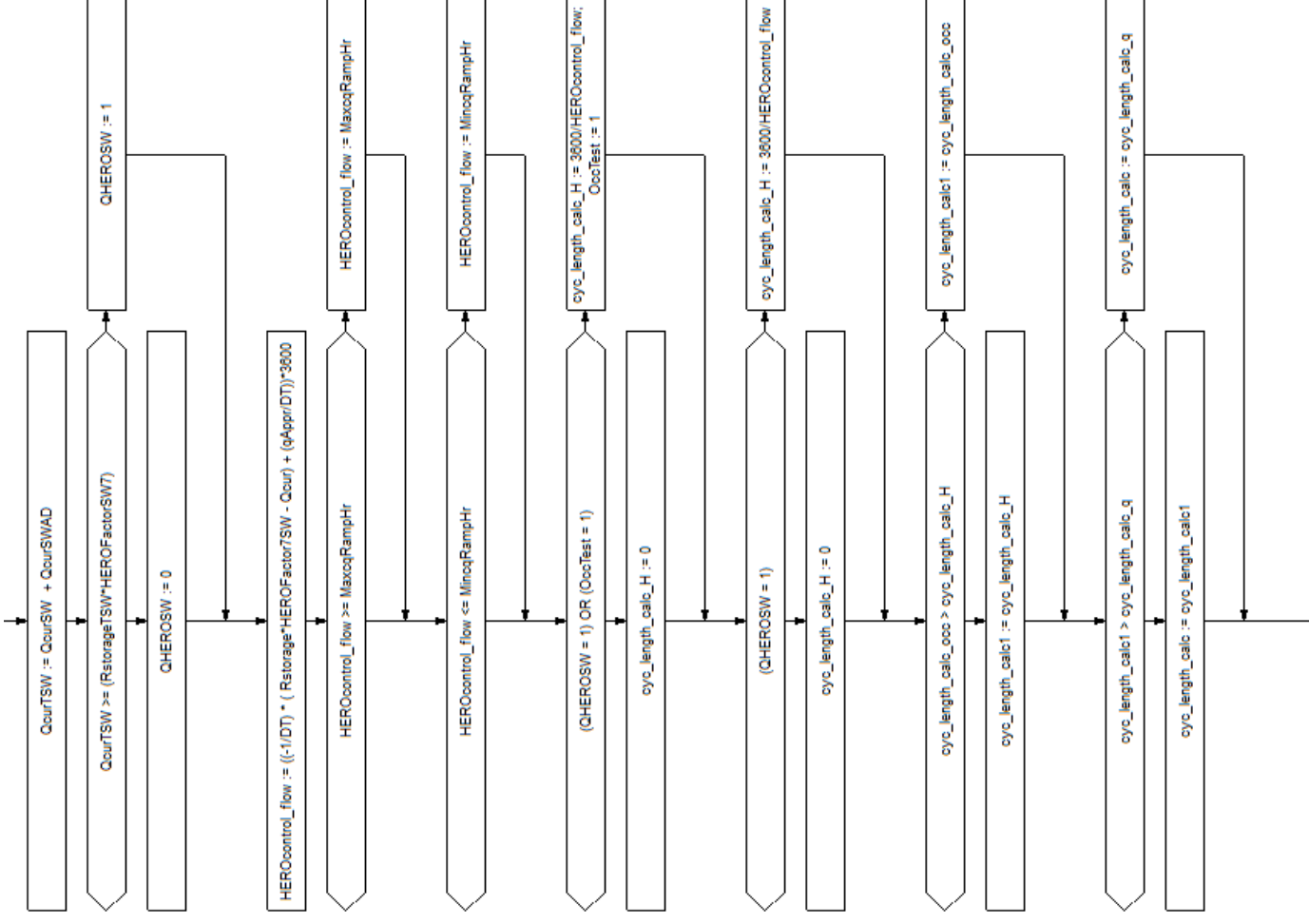
**VisVAP Flow Chart for the On-Ramp
Signal Controller in VISSIM:
HERO Algorithm**

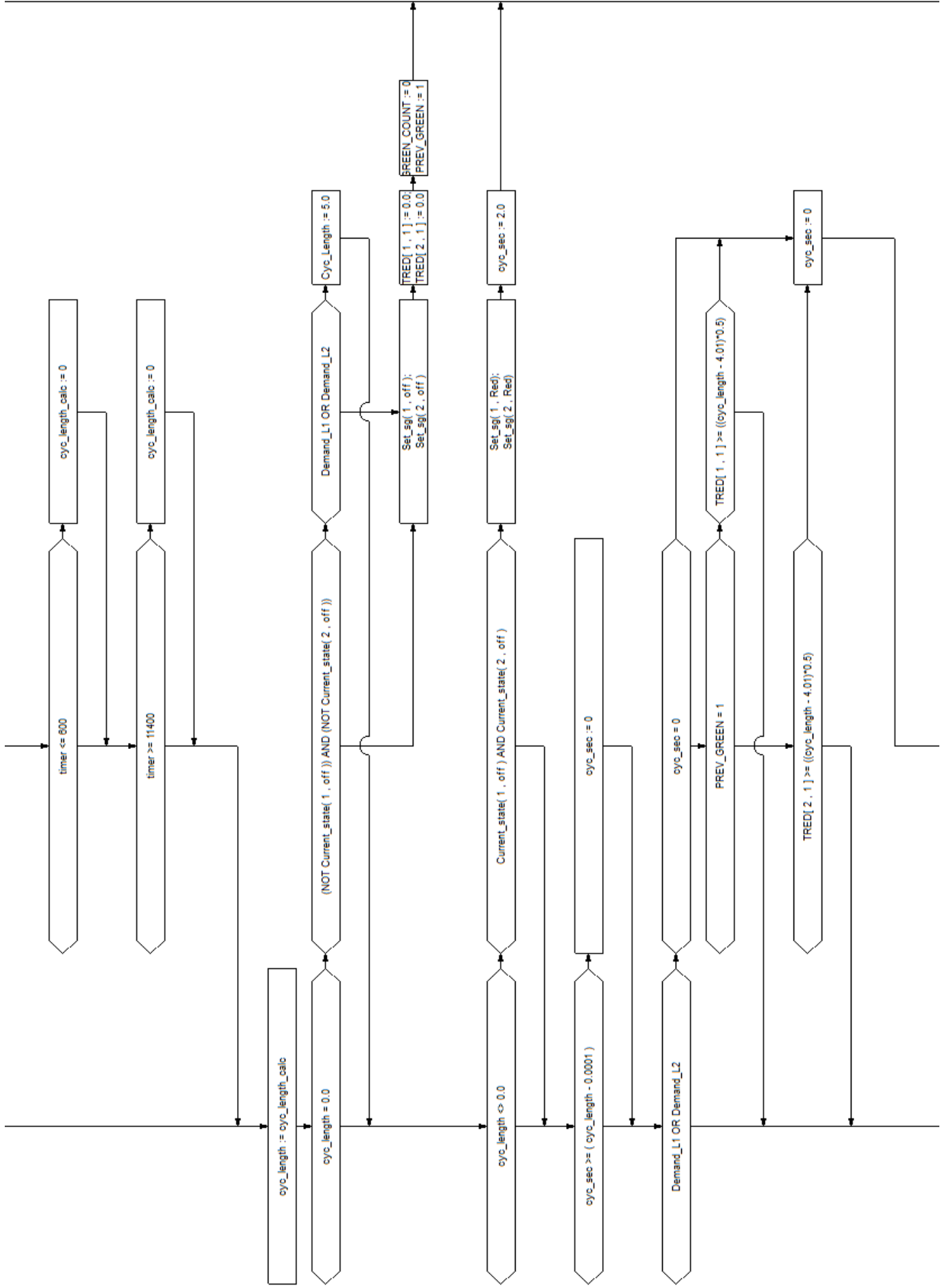


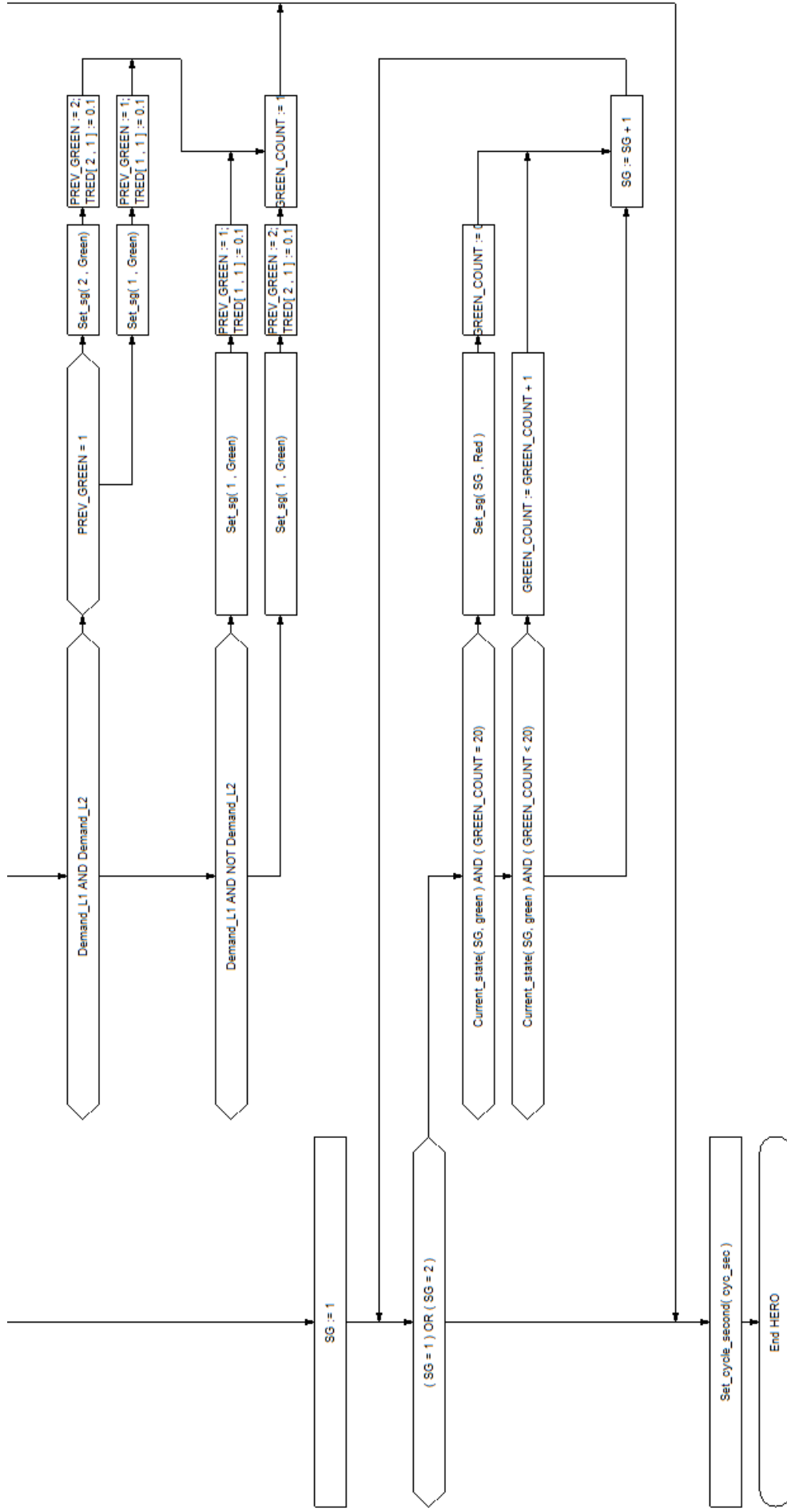












PARAMETERS, ARRAYS and EXPRESSIONS USED IN THE HERO ALGORITHM VisVAP FLOW CHART

PARAMETERS	Gen	Prog 1,...8	Comments
RDETAPP_L1	10		7 th St. Ramp detector approaching stop bar, lane 1.
RDETAPP_L2	11		7 th St. Ramp detector approaching stop bar, lane 2.
RDETDPEP_L1	12		7 th St. Ramp detector departing stop bar, lane 1.
RDETDPEP_L2	13		7 th St. Ramp detector departing stop bar, lane 2.
RDETDPEP_Ler1	14		7 th St. Ramp detector departing stop bar, lane 1 (Queue error correction).
RDETDPEP_Ler2	15		7 th St. Ramp detector departing stop bar, lane 2 (Queue error correction).
RDETDQ1	18		7 th St. Ramp queue detector lane 1.
RDETDQ2	19		7 th St. Ramp queue detector lane 2.
RDETDQer1	16		7 th St. Ramp queue detector lane 1 (For error correction).
RDETDQer2	17		7 th St. Ramp queue detector lane 2 (For error correction).
RDETDPEPSW_L1	151		SW Blvd. Ramp detector departing stop bar, lane 1.
RDETDPEPSW_L2	152		SW Blvd. Ramp detector departing stop bar, lane 2.
RDETDPEPSW_Ler1	153		SW Blvd. Ramp detector departing stop bar, lane 1 (Queue error correction).
RDETDPEPSW_Ler2	154		SW Blvd. Ramp detector departing stop bar, lane 2 (Queue error correction).
RDETDQSW1	161		SW Blvd. Ramp queue detector lane 1
RDETDQSW2	162		SW Blvd. Ramp queue detector lane 2
RDETDQSWer1	163		SW Blvd. Ramp queue detector lane 1 (Queue error correction).
RDETDQSWer2	164		SW Blvd. Ramp queue detector lane 2 (Queue error correction).
RDETDQSWer3	165		SW Blvd. Ramp queue detector lane 1 (Advanced queue error correction).

RDETQSWer4	166		SW Blvd. Ramp queue detector lane 2 (Advanced queue error correction).
RDETSWAD1	167		SW Blvd. Advanced queue detector lane 1.
RDETSWAD2	168		SW Blvd. Advanced queue detector lane 2.
RDETSWADer1	169		SW Blvd. Advanced queue detector lane 1. (Advanced queue error correction).
RDETSWADer2	170		SW Blvd. Advanced queue detector lane 2. (Advanced queue error correction).
MAX_DET	3		Number of mainline detectors at 7 th St. junction.
DT	20		Ramp metering control time step.
KR	70		Regulator parameter.
OCC_OPT	13		Desired occupancy.
OoutThreshold	11		Mainline occupancy threshold to activate ramp metering.
Rstorage	24		7 th St. Ramp storage (vehicles).
RstorageSW	42		SW Blvd. Ramp storage (Stop bar approach detector to queue detector.)
RstorageSWAD	28		SW Blvd. Ramp storage (Queue detector to advanced queue detector).
RstorageTSW	70		SW Blvd. Ramp storage (RstorageSW + RstorageSWAD)
HEROFactor7SW	0.5		Slave ramp (7 th St. on-ramp) queue creation percentage.
HEROFactorSW7	0.5		Master ramp (SW Blvd. on-ramp) queue activation percentage.
MaxcqRampHr	720		Maximum ramp metering vehicle flow per lane
MincqRampHr	200		Minimum ramp metering vehicle flow per lane.

ARRAYS	Dim2	Dim1	[1]	Comment
TRED	2	1	0	Time since signal 1 turned red. at 7 th St
TRED[2]			0	Time since signal 2 turned red. at 7 th St
DetNo	3	1	101	1 st mainline detector at 7 th St. at 7 th St
detNo[2]			102	2 nd mainline detector at 7 th St at 7 th St
detNo[3]			103	3 rd mainline detector at 7 th St at 7 th St

EXPRESSIONS	Contents	Comment
Demand_L1	Detection(RDETAPP_L1)	Detects vehicles waiting at the stop bar in Lane 1 at 7 th St.
Demand_L2	Detection(RDETAPP_L2)	Detects vehicles waiting at the stop bar in Lane 2 at 7 th St.

HERO

```

PROGRAM HERO_7TH; /
VAP_FREQUENCY 10;

CONST
    RDETAPP_L1 = 10,
    RDETAPP_L2 = 11,
    RDETDEP_L1 = 12,
    RDETDEP_L2 = 13,
    RDETDEP_Ler1 = 14,
    RDETDEP_Ler2 = 15,
    RDETQ1 = 18,
    RDETQ2 = 19,
    RDETQer1 = 16,
    RDETQer2 = 17,
    RDETDEPSW_L1 = 151,
    RDETDEPSW_L2 = 152,
    RDETDEPSW_Ler1 = 153,
    RDETDEPSW_Ler2 = 154,
    RDETQSW1 = 161,
    RDETQSW2 = 162,
    RDETQSWer1 = 163,
    RDETQSWer2 = 164,
    RDETQSWer3 = 165,
    RDETQSWer4 = 166,
    RDETQSWAD1 = 167,
    RDETQSWAD2 = 168,
    RDETQSWADer1 = 169,
    RDETQSWADer2 = 170,
    MAX_DET = 3,
    DT = 20,
    KR = 70,
    OCC_OPT = 13,
    OoutThreshold = 11,
    RStorage = 24,
    RStorageSW = 42,
    RStorageSWAD = 28,
    RstorageTSW = 70,
    SFactor = 0.75,
    HEROFactor7SW = 0.5,
    HEROFactorSW7 = 0.5,
    MaxcqRampHr = 720,
    MincqRampHr = 200;

/* ARRAYS */
ARRAY
    TRED[ 2, 1 ] = [[0], [0]],
    detNo[ 3, 1 ] = [[101], [102], [103]];

/* SUBROUTINES */

/* PARAMETERS DEPENDENT ON SCJ-PROGRAM */

/* EXPRESSIONS */
    Demand_L1 := Detection( RDETAPP_L1 );
    Demand_L2 := Detection( RDETAPP_L2 );

/* MAIN PROGRAM */
S00Z001:  IF int = 0 THEN
S01Z001:    int := 1;
S01Z002:    Set_sg( 1 , off); Set_sg( 2 , off);
S01Z003:    timer := 0.1; control_timer := 0.1;
S01Z004:    Queue_error := 0.1;

```



```

S01Z005:      Queue_errorSW := 0.1; Queue_errorSWAD := 0.1;
S01Z006:      QcurSW := 0; QcurSWAD := 0;
S01Z007:      cyc_sec := 0.1;
S01Z008:      cyc_length_calc := 0;
S01Z009:      TRED[ 1 , 1 ] := 0.0; TRED[ 2 , 1 ] := 0.0;
S01Z010:      GREEN_COUNT := 0;
S01Z011:      PREV_GREEN := 1; OccTest := 0;
S01Z012:      QHEROSW := 0 ; Qcur := 0;
S01Z013:      OoutPrev := 0;
S01Z014:      Oout100:=0; PcqRampHr := 0
ELSE
S00Z015:      timer := timer + 0.1; control_timer := control_timer + 0.1;
S00Z016:      cyc_sec := cyc_sec + 0.1;
S00Z017:      IF Current_state( 1 , red) THEN
S01Z017:          TRED[ 1 , 1 ] := TRED[ 1 , 1 ] + 0.1;
S00Z019:          IF Current_state( 2 , red) THEN
S01Z019:              TRED[ 2 , 1 ] := TRED[ 2 , 1 ] + 0.1;
S00Z021:              MdetNo := 1;
S00Z023:              IF MdetNo <= MAX_DET THEN
S01Z023:                  IF detNo[ MdetNo , 1 ] > 0 THEN
S02Z023:                      Oout := Oout + Occup_rate( detNo[ MdetNo , 1 ] );
S02Z024:                      MdetNo := MdetNo + 1;
                      GOTO S00Z023
                  ELSE
S00Z025:                      IF Detection( RDETQer1 ) OR Detection( RDETQer2 ) THEN
S01Z025:                          Queue_error := 0;
S00Z028:                          IF Detection( RDETQSWer1 ) OR Detection( RDETQSWer2 ) THEN
S01Z028:                              Queue_errorSW := 0;
S00Z031:                              IF Detection( RDETQSWAder1 ) OR Detection( RDETQSWAder2 )
THEN
S01Z031:                                  Queue_errorSWAD := 0;
S00Z034:                                  IF Queue_error >=10.99 THEN
S01Z034:                                      Queue_error := 0;
S01Z035:                                      Qerror_exit := (Front_ends( RDETDEP_Ler1 )) +
(Front_ends( RDETDEP_Ler2 ));
S01Z036:                                      Clear_front_ends( RDETDEP_Ler1 ) ; Clear_front_ends(
RDETDEP_Ler2 );
S01Z037:                                      Qerror_enter := Front_ends( RDETQer1 ) + Front_ends(
RDETQer2 );
S01Z038:                                      Clear_front_ends( RDETQer1 ); Clear_front_ends(
RDETQer2 );
S01Z039:                                      IF (Qerror_exit <= 0) AND (Qerror_enter <= 0) THEN
S02Z039:                                          Qcur := 0;
S00Z042:                                          IF Queue_errorSW >=12.99 THEN
S01Z042:                                              Queue_errorSW := 0;
S01Z043:                                              Qerror_exitSW := (Front_ends( RDETDEPSW_Ler1 )) +
(Front_ends( RDETDEPSW_Ler2 ));
S01Z044:                                              Clear_front_ends( RDETDEPSW_Ler1 ) ;
S01Z045:                                              Clear_front_ends( RDETDEPSW_Ler2 );
S01Z046:                                              Qerror_enterSW := Front_ends( RDETQSWer1 ) +
Front_ends( RDETQSWer2 );
S01Z047:                                              Clear_front_ends( RDETQSWer1 ); Clear_front_ends(
RDETQSWer2 );
S01Z048:                                              IF (Qerror_exitSW <= 0) AND (Qerror_enterSW <= 0)
THEN
S02Z048:                                                  QcurSW := 0;
S00Z049:                                                  IF Queue_errorSWAD >= 6.99 THEN
S01Z049:                                                      Queue_errorSWAD := 0;
S01Z050:                                                      Qerror_exitSWAD := (Front_ends( RDETQSWer3 ))
+ (Front_ends( RDETQSWer4 ));
S01Z051:                                                      Clear_front_ends( RDETQSWer3 );
S01Z052:                                                      Qerror_enterSWAD := Front_ends( RDETQSWAder1 )

```

```

+ Front_ends( RDETQSWADer2 );
S01Z053:
Clear_front_ends( RDETQSWADer2 );
S01Z054:
(Qerror_enterSWAD <= 0) THEN
S02Z054:
S00Z057:
S01Z057:
S01Z058:
S01Z059:
S01Z060:
Oout100);
S01Z061:
S02Z061:
S01Z063:
S02Z063:
S01Z065:
S01Z066:
S01Z067:
(OoutPrev < OoutThreshold)) AND (OccTest = 0) THEN
S02Z067:
S01Z070:
S01Z071:
S01Z072:
+ (Front_ends( RDETDEP_L2 ));
S01Z073:
Clear_front_ends( RDETDEP_L2 );
S01Z074:
S01Z075:
Front_ends( RDETQ2 );
S01Z076:
Clear_front_ends( RDETQ2 );
S01Z077:
S01Z078:
qRampHr);
S01Z079:
S02Z079:
S01Z081:
S02Z081:
S01Z083:
(Rstorage*SFactor - Qcur) + (qAppr/DT))*3600;
S01Z084:
MaxcqRampHr THEN
S02Z084:
MaxcqRampHr;
S01Z086:
MincqRampHr THEN
S02Z086:
MincqRampHr;
S01Z088:
3600/RampM_Rate_QueueHr;
S01Z089:
RDETQSW1 ) + Front_ends( RDETQSW2 ); qApprSWAD := Front_ends( RDETQSWAD1 ) +
Front_ends( RDETQSWAD2 );
S01Z090:
RDETDEPSW_L1 ) + Front_ends( RDETDEPSW_L2 );
S01Z091:
); Clear_front_ends( RDETQSW2 );
S01Z092:
); Clear_front_ends( RDETQSWAD2 );
S01Z093:
RDETDEPSW_L1 ); Clear_front_ends( RDETDEPSW_L2 );
S01Z094:

Clear_front_ends( RDETQSWADer1 );
IF (Qerror_exitSWAD <= 0) AND
QcurSWAD := 0;
IF control_timer >= ( DT - 0.0001 ) THEN
control_timer := 0;
Oout:= Oout/MAX_DET/(DT*10);
Oout100 := Oout*100;
cqRampHr := PcqRampHr + KR*(OCC_OPT -
IF cqRampHr >= MaxcqRampHr THEN
cqRampHr := MaxcqRampHr;
IF cqRampHr <= MincqRampHr THEN
cqRampHr := MincqRampHr;
cyc_length_calc_occ := 3600/cqRampHr;
PcqRampHr := cqRampHr;
IF ((Oout100 > OoutThreshold) OR
THEN
cyc_length_calc_occ := 0;
OoutPrev := Oout100;
Oout := 0;
qRamp := (Front_ends( RDETDEP_L1 ))
Clear_front_ends( RDETDEP_L1 );
qRampHr := 3600*qRamp/DT;
qAppr := Front_ends( RDETQ1 ) +
Clear_front_ends( RDETQ1 );
qApprHr := 3600 * qAppr/DT;
Qcur := Qcur+(DT/3600)*(qApprHr -
IF Qcur < 0 THEN
Qcur := 0;
IF Qcur > (Rstorage + 1) THEN
Qcur := (Rstorage + 1);
RampM_Rate_QueueHr := ((-1/DT) *
IF RampM_Rate_QueueHr >=
RampM_Rate_QueueHr :=
IF RampM_Rate_QueueHr <=
RampM_Rate_QueueHr :=
cyc_length_calc_q :=
qApprSW := Front_ends(
qRampSW := Front_ends(
Clear_front_ends( RDETQSW1
Clear_front_ends( RDETQSWAD1
Clear_front_ends(
qApprHrSW :=

```

```

3600*qApprSW/DT; qApprHrSWAD := 3600*qApprSWAD/DT;
S01Z095: qRampHrSW :=
3600*qRampSW/DT;
S01Z096: QcurSW := QcurSW
+(DT/3600)*(qApprHrSW - qRampHrSW); QcurSWAD := QcurSWAD +(DT/3600)*(qApprHrSWAD -
qApprHrSW );
S01Z097: IF QcurSW <= 0 THEN
S02Z097: QcurSW := 0;
S01Z099: IF QcurSW >= (RstorageSW +
1) THEN
S02Z099: QcurSW := (RstorageSW +
1);
S01Z101: IF QcurSWAD <= 0 THEN
S02Z101: QcurSWAD := 0;
S01Z103: IF QcurSWAD >=
(RstorageSWAD + 1) THEN
S02Z103: QcurSWAD :=
(RstorageSWAD + 1);
S01Z105: QcurTSW := QcurSW +
QcurSWAD;
S01Z106: IF QcurTSW >=
(RstorageTSW*HEROFactorSW7) THEN
S02Z106: QHEROSW := 1;
S01Z109: HEROcontrol_flow
:= ((-1/DT) * ( Rstorage*HEROFactor7SW - Qcur) + (qAppr/DT))*3600;
S01Z110: IF
HEROcontrol_flow >= MaxcqRampHr THEN
S02Z110: HEROcontrol_flow
:= MaxcqRampHr;
S01Z112: IF
HEROcontrol_flow <= MincqRampHr THEN
S02Z112:
HEROcontrol_flow := MincqRampHr;
S01Z114: IF (QHEROSW =
1) OR (OccTest = 1) THEN
S02Z114:
cyc_length_calc_H := 3600/HEROcontrol_flow; OccTest := 1;
S01Z117: IF (QHEROSW
= 1) THEN
S02Z117:
cyc_length_calc_H := 3600/HEROcontrol_flow;
S01Z120: IF
cyc_length_calc_occ > cyc_length_calc_H THEN
S02Z120:
cyc_length_calc1 := cyc_length_calc_occ;
S01Z123: IF
cyc_length_calc1 > cyc_length_calc_q THEN
S02Z123:
cyc_length_calc := cyc_length_calc_q;
S01Z127: IF
timer <= 600 THEN
S02Z127:
cyc_length_calc := 0;
S01Z129: IF
timer >= 11400 THEN
S02Z129:
cyc_length_calc := 0;
S00Z132:
cyc_length := cyc_length_calc;
S00Z133: IF
cyc_length = 0.0 THEN
S01Z133:
IF (NOT Current_state( 1 , off )) AND (NOT Current_state( 2 , off )) THEN

```

```

S02Z133:
  IF Demand_L1 OR Demand_L2 THEN
S03Z133:
    Cyc_Length := 5.0;
S00Z137:
    IF cyc_length <> 0.0 THEN
S01Z137:
      IF Current_state( 1 , off ) AND Current_state( 2 , off ) THEN
S02Z137:
        Set_sg( 1 , Red); Set_sg( 2 , Red);
S03Z137:
        cyc_sec := 2.0

      ELSE
S00Z139:
        IF cyc_sec >= ( cyc_length - 0.0001 ) THEN
S01Z139:
          cyc_sec := 0;
S00Z141:
          IF Demand_L1 OR Demand_L2 THEN
S01Z141:
            IF cyc_sec = 0 THEN
S03Z144:
              cyc_sec := 0;
S01Z148:
              IF Demand_L1 AND Demand_L2 THEN
S02Z148:
                IF PREV_GREEN = 1 THEN
S03Z148:
                  Set_sg( 2 , Green);
S04Z148:
                  PREV_GREEN := 2; TRED[ 2 , 1 ] := 0.1;
S04Z152:
                  GREEN_COUNT := 1

                ELSE
S03Z149:
                  Set_sg( 1 , Green);
S04Z149:
                  PREV_GREEN := 1; TRED[ 1 , 1 ] := 0.1;
                  GOTO S04Z152

                END
              ELSE
S01Z151:
                IF Demand_L1 AND NOT Demand_L2 THEN
S02Z151:
                  Set_sg( 1 , Green);
S03Z151:
                  PREV_GREEN := 1; TRED[ 1 , 1 ] := 0.1;
                  GOTO S04Z152

                ELSE
S02Z152:
                  Set_sg( 1 , Green);
S03Z152:
                  PREV_GREEN := 2; TRED[ 2 , 1 ] := 0.1;
                  GOTO S04Z152

```

```

        END
    END
ELSE
S01Z142:   IF PREV_GREEN = 1 THEN
S02Z142:   IF TRED[ 1 , 1 ] >= ((cyc_length - 4.01)*0.5) THEN
        GOTO S03Z144
    ELSE
S00Z153:   SG := 1;
S00Z155:   IF ( SG = 1 ) OR ( SG = 2 ) THEN
S01Z156:   IF Current_state( SG, green ) AND ( GREEN_COUNT = 20) THEN
S02Z156:   Set_sg( SG , Red );
S03Z156:   GREEN_COUNT := 0;
S04Z159:   SG := SG + 1;
        GOTO S00Z155
    ELSE
S01Z157:   IF Current_state( SG, green ) AND ( GREEN_COUNT < 20) THEN
S02Z157:   GREEN_COUNT := GREEN_COUNT + 1;
        GOTO S04Z159
    ELSE
        GOTO S04Z159
    END
    END
    END
    END
    END
ELSE
S01Z144:   IF TRED[ 2 , 1 ] >= ((cyc_length - 4.01)*0.5) THEN
        GOTO S03Z144
    ELSE
        GOTO S00Z153
    END
    END
    END
END
END

```



```

S01Z118:
cyc_length_calc_H := 0;

S01Z120
GOTO
END
ELSE

S01Z115:
cyc_length_calc_H := 0;

GOTO S01Z117
END
ELSE
GOTO S01Z114
END
ELSE
GOTO S01Z112
END
ELSE
QHEROSW := 0;
GOTO S01Z109
END
ELSE
GOTO S01Z105
END
ELSE
GOTO S01Z103
END
ELSE
GOTO S01Z101
END
ELSE
GOTO S01Z099
END
ELSE
GOTO S01Z088
END
ELSE
GOTO S01Z086
END
ELSE
GOTO S01Z083
END
ELSE
GOTO S01Z081
END
ELSE
OccTest := 1;
GOTO S01Z070
END
ELSE
GOTO S01Z065
END
ELSE
GOTO S01Z063
END
ELSE
GOTO S00Z132
END
ELSE
GOTO S00Z057
END
ELSE
GOTO S00Z057
END

```

```

                                ELSE
                                GOTO S00Z049
                                END
                                ELSE
                                GOTO S00Z049
                                END
                                ELSE
                                GOTO S00Z042
                                END
                                ELSE
                                GOTO S00Z042
                                END
                                ELSE
S00Z032:      Queue_errorSWAD := Queue_errorSWAD + 0.1;
                                GOTO S00Z034
                                END
                                ELSE
S00Z029:      Queue_errorSW := Queue_errorSW + 0.1;
                                GOTO S00Z031
                                END
                                ELSE
S00Z026:      Queue_error := Queue_error + 0.1;
                                GOTO S00Z028
                                END
                                END
                                ELSE
                                GOTO S00Z025
                                END
                                ELSE
                                GOTO S00Z021
                                END
                                ELSE
                                GOTO S00Z019
                                END
                                END;
S00Z161:      Set_cycle_second( cyc_sec )
PROG_ENDE:      .
/*-----*/

```